



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**TUGAS AKHIR - TE 141599**

**IMPLEMENTASI DAN ANALISIS KINERJA *NETWORK CODING*  
PADA SISTEM KOMUNIKASI NIRKABEL MODULASI QAM  
MENGUNAKAN *WIRELESS OPEN-ACCESS RESEARCH*  
*PLATFORM***

Prastiyanto  
NRP 2213106003

Dosen Pembimbing  
Dr. Ir. Suwadi, M.T.  
Dr. Ir. Wirawan, DEA.

JURUSAN TEKNIK ELEKTRO  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2016



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**FINAL PROJECT - TE 141599**

**IMPLEMENTATION AND PERFORMANCE ANALYSIS NETWORK  
CODING IN QAM MODULATION WIRELESS COMMUNICATION  
USING WIRELESS OPEN-ACCESS RESEARCH PLATFORM**

Prastiyanto  
NRP 2213106003

Supervisors  
Dr. Ir. Suwadi, M.T.  
Dr. Ir. Wirawan, DEA.

DEPARTMENT OF ELECTRICAL ENGINEERING  
Faculty of Industrial Technical  
Institut Teknologi Sepuluh Nopember  
Surabaya 2016




**IMPLEMENTASI DAN ANALISIS KINERJA NETWORK  
CODING PADA SISTEM KOMUNIKASI NIRKABEL  
MODULASI QAM MENGGUNAKAN WIRELESS OPEN-ACCESS  
RESEARCH PLATFORM**

**TUGAS AKHIR**

**Diajukan untuk Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Sarjana Teknik  
Pada  
Bidang Studi Teknik Telekomunikasi dan Multimedia  
Jurusan Teknik Elektro  
Institut Teknologi Sepuluh Nopember**

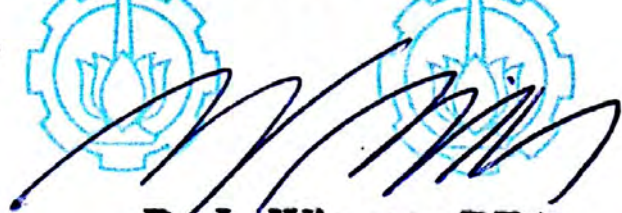
**Menyetujui:**

**Dosen Pembimbing I**

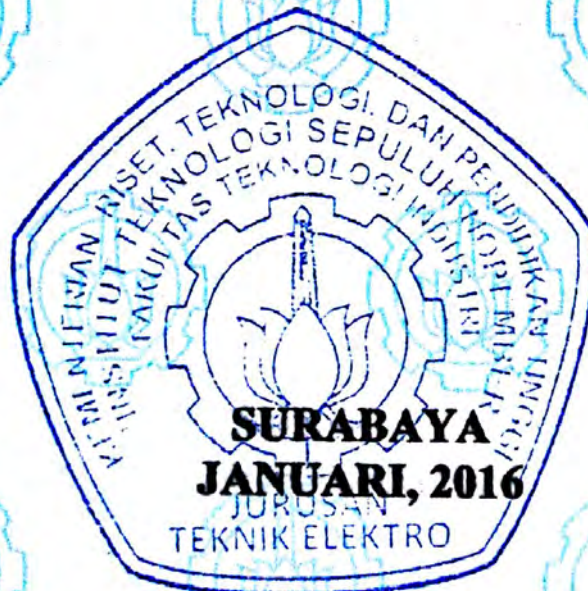


**Dr. Ir. Suwadi, M.T.  
NIP. 19680818 1993031 002**

**Dosen Pembimbing II**



**Dr. Ir. Wirawan, DEA.  
NIP. 19631109 1989031 011**





# **Implementasi dan Analisis Kinerja *Network Coding* pada Sistem Komunikasi Nirkabel Modulasi QAM Menggunakan *Wireless Open-Access Research Platform***

**Nama** : Prastiyanto  
**Pembimbing** : Dr. Ir. Suwadi, M.T.  
Dr. Ir. Wirawan, DEA.

## **ABSTRAK**

Sistem komunikasi nirkabel merupakan bagian dari teknologi telekomunikasi yang kini menjadi solusi bagi setiap individu yang memiliki mobilitas yang cukup tinggi. Kebutuhan akan akses data yang cepat namun tidak berbanding lurus dengan ketersediaan *Bandwidth* yang terbatas dan *Throughput* yang relatif rendah. Berdasarkan teori yang sudah ada mengenai *Network Coding* dan Modulasi *Quadrature Amplitude Modulation* (QAM), dibuat sebuah simulasi dengan konsep Sistem Komunikasi Nirkabel Modulasi QAM dengan *Network Coding*. Simulasi tersebut bertujuan untuk membuktikan konsep yang diteliti dapat disimulasikan dan direalisasikan. Pada simulasi, sebuah sistem modulasi QAM dengan skema *network coding* dapat memberikan perbaikan nilai *throughput* mencapai 31,8 % dari sistem modulasi QAM tanpa skema *network coding*. Setelah dapat disimulasikan, konsep tersebut diimplementasikan pada perangkat *Wireless Open-Access Research Platform* (WARP) yang merupakan salah satu jenis *Software Defined Radio* (SDR), yang memungkinkan untuk pengaplikasian konsep Sistem Komunikasi Nirkabel Modulasi QAM dengan *Network Coding*. Perangkat WARP tersebut akan digunakan sebagai model pemancar dan penerima dari dua sumber dan satu *relay* untuk pengukuran kualitas unjuk kerja sistem Modulasi QAM dengan *Network Coding*. Pengukuran dilakukan pada kondisi lingkungan *Indoor* dan *Outdoor* dengan beberapa perbedaan jarak antar dua sumber. Hasil pengukuran implementasi menunjukkan sebuah sistem modulasi QAM dengan skema *network coding* dapat memberikan perbaikan nilai *throughput* mencapai 19,67 % dari sebuah sistem modulasi QAM tanpa skema *network coding*.

Kata Kunci–*Network Coding*, QAM, *Throughput*, WARP.

# **Implementation and Performance Analysis Network Coding in QAM Modulation Wireless Communication Using Wireless Open-Access Research Platform**

**Name** : Prastiyanto  
**Supervisors** : Dr. Ir. Suwadi, M.T.  
Dr. Ir. Wirawan, DEA.

## **ABSTRACT**

Wireless communication system is part of a telecommunication technology that now is a solution for every individual who has a fairly high mobility. The need for fast data access but not directly proportional to the limited availability of bandwidth and throughput is relatively low. Based on existing theories on Network Coding and Quadrature Amplitude Modulation (QAM), created a simulation with the concept of QAM Modulation Wireless Communication System with Network Coding. The simulation aims to prove the concept under study can be simulated and realized. In the simulation, a QAM modulation system with network coding scheme can deliver improved throughput value reached 31.8% of the QAM modulation systems without network coding scheme. Once can be simulated, the concept is implemented in the Wireless Open-Access Research Platform (WARP) which is one type of Software Defined Radio (SDR), which allows for the application of the concept of QAM Modulation Wireless Communication System with Network Coding. WARP device will be used as a transmitter and receiver models from two sources and one relay for measuring the quality of the performance of the QAM modulation system with Network Coding. Measurements carried out on the environmental conditions Indoor and Outdoor with some differences in distance between the two sources. Implementation of the measurement result shows a QAM modulation system with network coding scheme can deliver improved throughput reached 19.67% from a QAM modulation systems without network coding scheme.

*Keyword—Network Coding, QAM, Throughput, WARP.*

## KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Alhamdulillah segala puji syukur kami panjatkan kehadirat Allah SWT, karena atas limpahan rahmat dan hidayah-Nya penulis dapat menyelesaikan Tugas Akhir yang berjudul:

**“Implementasi dan Analisis Kinerja *Network Coding* pada Sistem Komunikasi Nirkabel Modulasi QAM Menggunakan *Wireless Open-Access Research Platform*”**

Tugas akhir ini merupakan salah satu syarat yang harus dipenuhi dalam menyelesaikan program studi strata 1 pada jurusan teknik elektro, fakultas teknologi industri, Institut Teknologi Sepuluh Nopember. Dalam proses pengerjaan penelitian ini penulis banyak mendapat saran, dorongan, bimbingan serta keterangan-keterangan dari berbagai pihak. Oleh karena itu dengan segala hormat dan kerendahan hati penulis mengucapkan terima kasih kepada:

1. Kedua orang tua penulis yang berada di Jakarta, yang senantiasa mendoakan, memberikan semangat, perhatian, motivasi, dukungan moril, dan dukungan materil.
2. Bapak Dr. Ir. Suwadi, M.T. dan Bapak Dr. Ir. Wirawan, DEA. yang selalu memberikan dukungan, semangat, motivasi, bantuan teknis dan non teknis dari awal pengajuan Tugas Akhir sampai pengajuan maju sidang Tugas Akhir.
3. Teman-teman satu tim WARP, yang sama-sama berjuang sepanjang waktu pengerjaan Tugas Akhir, saling menyemangati, saling mengingatkan, dan saling mendoakan.
4. Teman-teman seperjuangan Telkom Ceria yang selalu memberikan semangat dan hiburan ketika jenuh melanda.
5. Anak-anak Kontrakan Bujang yang telah menemani selama berjuang menempuh pendidikan di kota Surabaya.

Dalam proses perbaikan dan pengembangan kritik dan saran terhadap penelitian ini sangat dibutuhkan untuk mengetahui kekurangan dan kebutuhannya. Akhir kata penulis ucapkan terima kasih.

Wassalamu'alaikum Wr. Wb.

Surabaya, 20 Januari 2016

Penulis



# DAFTAR ISI

	HALAMAN
ABSTRAK .....	i
ABSTRACT .....	iii
KATA PENGANTAR .....	v
DAFTAR ISI .....	vii
DAFTAR GAMBAR .....	xi
DAFTAR TABEL .....	xv
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Perumusan Masalah .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan Penelitian .....	2
1.5 Metodologi Penelitian .....	3
1.5.1 Studi Literatur .....	3
1.5.2 Pembuatan Simulasi Menggunakan <i>Software</i> Matlab .....	3
1.5.3 Implementasi ke Modul WARP .....	3
1.5.4 Uji Coba dan Perbaikan .....	4
1.5.5 Pengukuran .....	4
1.5.6 Analisis .....	4
1.6 Sistematika .....	4
1.7 Manfaat .....	5
BAB II TEORI PENUNJANG .....	7
2.1 <i>Quadrature Amplitude Modulation</i> (QAM) .....	7
2.1.1 Penentuan Amplitudo Sinyal .....	8
2.1.2 <i>Gray Mapping</i> ( <i>Gray Coding</i> ) [6] .....	9
2.1.3 Probabilitas <i>Bit Error</i> pada M-QAM [6] .....	12
2.2 <i>Network Coding</i> .....	14
2.2.1 Skema <i>Non-Network Coding</i> [7] .....	15
2.2.2 Skema <i>Straightformard Network Coding</i> .....	15
2.2.3 Pertimbangan <i>Noise</i> dan <i>Throughput</i> [7] .....	16
2.3 <i>Wireless Open-Access Research Platform</i> (WARP) .....	19

2.3.1	Arsitektur <i>Platform</i> [3] .....	20
2.3.2	WARP FPGA <i>Board</i> v2.....	21
2.3.3	WARP <i>Radio Board</i> v1.4 .....	22
2.3.4	WARPLab.....	23
2.3.5	Spesifikasi WARPLab [10] .....	27
2.4	<i>Bit Error Rate</i> (BER) dan <i>Throughput</i> .....	28
BAB III PERANCANGAN DAN IMPLEMENTASI SISTEM.....		31
3.1	Perancangan Simulasi Sistem pada Matlab .....	31
3.2	Integrasi PC dengan Modul WARP .....	34
3.2.1	Konfigurasi IP pada PC .....	35
3.2.2	Tes Koneksi antara PC dan Modul WARP .....	36
3.3	Implementasi Sistem pada Modul WARP.....	38
3.3.1	Parameter Sistem .....	40
3.3.2	Desain <i>Frame</i> .....	40
3.3.3	Perancangan Sistem di Pemancar .....	40
3.3.4	Perancangan Sistem di WARP Transmit .....	46
3.3.5	Perancangan Sistem di Penerima .....	46
3.3.6	Perancangan Sistem <i>Network Coding</i> .....	47
3.4	Skenario Pengukuran .....	47
3.4.1	Lingkungan Dalam Ruangan ( <i>Indoor</i> ).....	47
3.4.2	Lingkungan Luar Ruangan ( <i>Outdoor</i> ) .....	48
BAB IV HASIL DAN ANALISIS DATA .....		51
4.1	Hasil Simulasi Sistem Modulasi M-QAM dengan Network Coding 51	
4.1.1	Hasil Simulasi <i>Bit Error Rate</i> (BER).....	51
4.1.2	Hasil Simulasi <i>Throughput</i> .....	56
4.2	Analisis Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	62
4.2.1	Analisis Sistem pada Modulasi 4-QAM .....	63
4.2.2	Analisis Sistem pada Modulasi 8-QAM .....	66
4.2.3	Analisis Sistem pada Modulasi 16-QAM .....	70
4.2.4	Analisis Sistem pada Modulasi 32-QAM .....	73
4.2.5	Analisis Sistem pada Modulasi 64-QAM .....	77



4.3	Analisis Sistem <i>Network Coding</i> dengan Variasi Jarak .....	80
4.3.1	Analisis Sistem pada Modulasi 4-QAM .....	81
4.3.2	Analisis Sistem pada Modulasi 8-QAM .....	85
4.3.3	Analisis Sistem pada Modulasi 16-QAM .....	89
4.3.4	Analisis Sistem pada Modulasi 32-QAM .....	93
4.3.5	Analisis Sistem pada Modulasi 64-QAM .....	97
4.4	Analisis Sistem <i>Network Coding</i> pada Modulasi M-QAM .....	100
4.4.1	Analisis Berdasarkan <i>Bit Error Rate</i> .....	101
4.4.2	Analisis Berdasarkan <i>Throughput</i> .....	105
BAB V PENUTUP .....		109
5.1	Kesimpulan .....	109
5.2	Saran .....	110
DAFTAR PUSTAKA .....		111
LAMPIRAN A LEMBAR PENGESAHAN PROPOSAL TA .....		113
LAMPIRAN B <i>LISTING PROGRAM</i> .....		115
LAMPIRAN C DOKUMENTASI .....		133

## DAFTAR GAMBAR

Gambar 2.1 Blok Diagram Pemancar QAM [4].....	8
Gambar 2.2 Blok Diagram Penerima QAM [4] .....	8
Gambar 2.3 Konstelasi Sinyal 4-QAM .....	10
Gambar 2.4 Konstelasi Sinyal 8-QAM .....	10
Gambar 2.5 Konstelasi Sinyal 16-QAM.....	10
Gambar 2.6 Konstelasi Sinyal 64-QAM .....	11
Gambar 2.7 Diagram Ruang Sinyal untuk Rectangular QAM .....	11
Gambar 2.8 Dua Konstelasi Sinyal QAM dengan $M = 4$ .....	13
Gambar 2.9 Empat Konstelasi Sinyal untuk 8-QAM .....	13
Gambar 2.10 Grafik Probabilitas <i>Error</i> Simbol M-QAM.....	14
Gambar 2.11 Skema <i>Tradisional Non-Network Coding</i> [7].....	15
Gambar 2.12 Skema <i>Straightforward Network Coding</i> [7].....	15
Gambar 2.13 Diagram Proses Skema <i>Straightforward Network Coding</i> [8].....	16
Gambar 2.14 Kurva BER QPSK, PNC, dan SNC.....	18
Gambar 2.16 Komponen-komponen <i>Platform</i> .....	20
Gambar 2.17 Arsitektur <i>Hardware</i> WARP FPGA [10] .....	21
Gambar 2.18 Bentuk Fisik WARP FPGA <i>Board</i> v2 [10] .....	22
Gambar 2.19 Bentuk Fisik WARP <i>Radio Board</i> v1.4 [11].....	23
Gambar 2.20 Arsitektur <i>Radio Board</i> v1.4 [11].....	23
Gambar 2.21 <i>WARPLab Framework</i> [9] .....	24
Gambar 2.22 Desain Arsitektur FPGA WARPLab [11] .....	25
Gambar 3.1 Blok Diagram Proses Simulasi Sistem Komunikasi M-QAM Traditional.....	31
Gambar 3.2 Blok Diagram Proses Simulasi Sistem Komunikasi M-QAM dengan Network Coding .....	32
Gambar 3.3 Hasil Simulasi BER 4 QAM <i>Relay</i> dan 4 QAM <i>Network</i> <i>Coding</i> .....	34
Gambar 3.4 Konfigurasi Integrasi PC dengan Modul WARP .....	35
Gambar 3.5 Pengaturan IP pada PC .....	35
Gambar 3.6 Konfigurasi IP ARP pada CMD .....	36
Gambar 3.7 Cek Koneksi Melalui <i>Command Prompt</i> .....	37
Gambar 3.8 Hasil dari <i>setup</i> WARP pada Matlab.....	38
Gambar 3.9 Hasil dari cek koneksi pada Matlab.....	38
Gambar 3.10 Blok Diagram Proses Kerja pada Pengirim .....	39
Gambar 3.11 Blok Diagram Proses Kerja pada Penerima.....	39



Gambar 3.12 Struktur <i>Frame</i> Implementasi Sistem pada WARP .....	40
Gambar 3.13 Bit Informasi yang Dibangkitkan.....	41
Gambar 3.14 Konstelasi Simbol pada Level Modulasi 4-QAM.....	42
Gambar 3.15 Konstelasi Simbol pada Level Modulasi 8-QAM.....	42
Gambar 3.16 Konstelasi Simbol pada Level Modulasi 16-QAM .....	43
Gambar 3.17 Konstelasi Simbol pada Level Modulasi 32-QAM.....	43
Gambar 3.18 Konstelasi Simbol pada Level Modulasi 64-QAM.....	44
Gambar 3.19 Susunan Pilot dan Simbol .....	44
Gambar 3.20 Respon <i>Impuls</i> Filter <i>Squared Root Raised Cosine</i> .....	45
Gambar 3.21 Konfigurasi Letak dan Jarak Tiap <i>Node</i> WARP <i>Indoor</i> ..	48
Gambar 3.22 Konfigurasi Letak dan Jarak Tiap <i>Node</i> WARP <i>Outdoor</i>	49
Gambar 4.1 Hasil Simulasi BER pada Modulasi 4-QAM .....	51
Gambar 4.2 Hasil Simulasi BER pada Modulasi 8-QAM .....	52
Gambar 4.3 Hasil Simulasi BER pada Modulasi 16-QAM .....	53
Gambar 4.4 Hasil Simulasi BER pada Modulasi 32-QAM .....	54
Gambar 4.5 Hasil Simulasi BER pada Modulasi 64-QAM .....	55
Gambar 4.6 Hasil Simulasi <i>Throughput</i> pada Modulasi 4-QAM .....	56
Gambar 4.7 Hasil Simulasi <i>Throughput</i> pada Modulasi 8-QAM .....	57
Gambar 4.8 Hasil Simulasi <i>Throughput</i> pada Modulasi 16-QAM .....	58
Gambar 4.9 Hasil Simulasi <i>Throughput</i> pada Modulasi 32-QAM .....	60
Gambar 4.10 Hasil Simulasi <i>Throughput</i> pada Modulasi 64-QAM .....	61
Gambar 4.11 Kurva BER 4-QAM <i>Indoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	63
Gambar 4.12 Kurva BER 4-QAM <i>Outdoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	63
Gambar 4.13 Kurva <i>Throughput</i> 4-QAM <i>Indoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	64
Gambar 4.14 Kurva <i>Throughput</i> 4-QAM <i>Outdoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	65
Gambar 4.15 Kurva BER 8-QAM <i>Indoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	66
Gambar 4.16 Kurva BER 8-QAM <i>Outdoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	67
Gambar 4.17 Kurva <i>Throughput</i> 8-QAM <i>Indoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	68
Gambar 4.18 Kurva <i>Throughput</i> 8-QAM <i>Outdoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	68

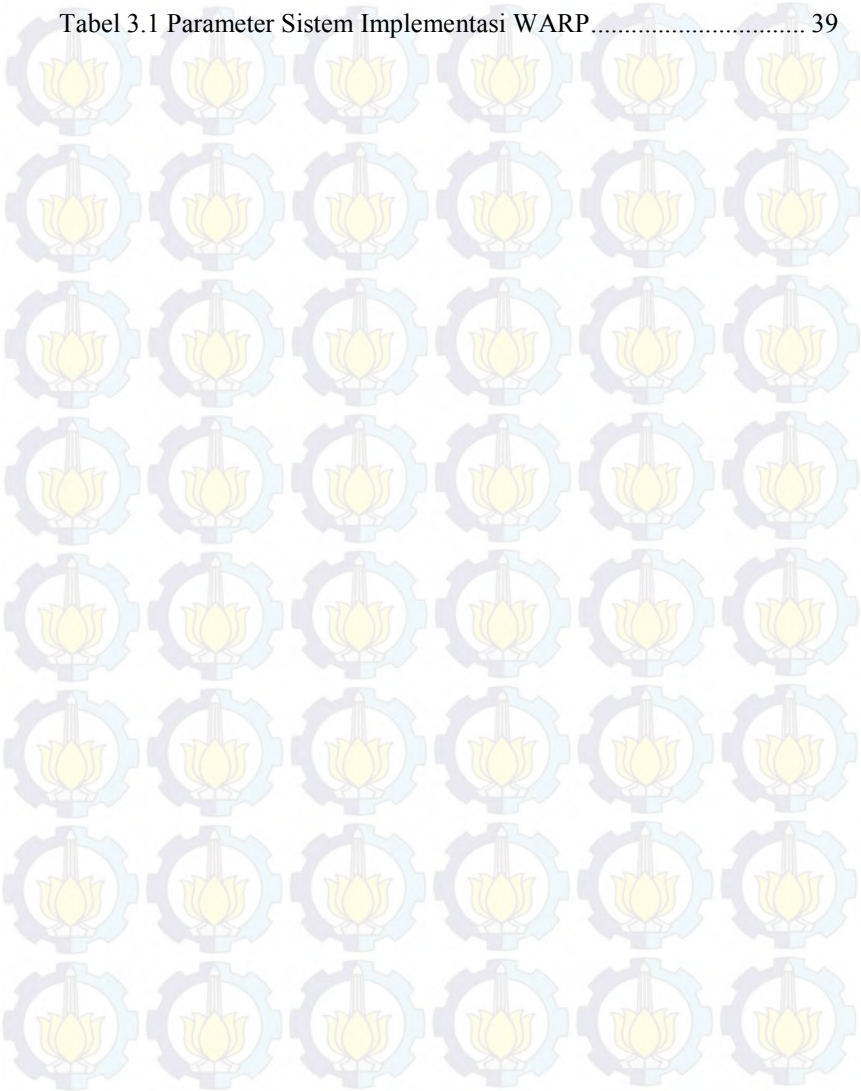
Gambar 4.19 Kurva BER 16-QAM <i>Indoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	70
Gambar 4.20 Kurva BER 16-QAM <i>Outdoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	70
Gambar 4.21 Kurva <i>Throughput</i> 16-QAM <i>Indoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	71
Gambar 4.22 Kurva <i>Throughput</i> 16-QAM <i>Outdoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	72
Gambar 4.23 Kurva BER 32-QAM <i>Indoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	73
Gambar 4.24 Kurva BER 32-QAM <i>Outdoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	74
Gambar 4.25 Kurva <i>Throughput</i> 32-QAM <i>Indoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	75
Gambar 4.26 Kurva <i>Throughput</i> 32-QAM <i>Outdoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	75
Gambar 4.27 Kurva BER 64-QAM <i>Indoor</i> dari Sistem <i>Network Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	77
Gambar 4.28 Kurva BER 64-QAM <i>Outdoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	77
Gambar 4.29 Kurva <i>Throughput</i> 64-QAM <i>Indoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	78
Gambar 4.30 Kurva <i>Throughput</i> 64-QAM <i>Indoor</i> dari Sistem <i>Network</i> <i>Coding</i> , Tanpa <i>Network Coding</i> , dan <i>Point to Point</i> .....	79
Gambar 4.31 Kurva BER <i>Network Coding</i> 4-QAM Terhadap Variasi Jarak .....	81
Gambar 4.32 Kurva <i>Throughput Network Coding</i> 4-QAM Terhadap Variasi Jarak.....	83
Gambar 4.33 Kurva BER <i>Network Coding</i> 8-QAM Terhadap Variasi Jarak .....	85
Gambar 4.34 Kurva <i>Throughput Network Coding</i> 8-QAM Terhadap Variasi Jarak.....	87
Gambar 4.35 Kurva BER <i>Network Coding</i> 16-QAM Terhadap Variasi Jarak .....	89
Gambar 4.36 Kurva <i>Throughput Network Coding</i> 16-QAM Terhadap Variasi Jarak.....	91
Gambar 4.37 Kurva BER <i>Network Coding</i> 32-QAM Terhadap Variasi Jarak .....	93



Gambar 4.38 Kurva <i>Throughput Network Coding</i> 32-QAM Terhadap Variasi Jarak .....	95
Gambar 4.39 Kurva BER <i>Network Coding</i> 64-QAM Terhadap Variasi Jarak .....	97
Gambar 4.40 Kurva <i>Throughput Network Coding</i> 64-QAM Terhadap Variasi Jarak .....	99
Gambar 4.41 Kurva BER <i>Network Coding Indoor</i> pada Modulasi M-QAM.....	101
Gambar 4.42 Kurva BER <i>Network Coding Outdoor</i> pada Modulasi M-QAM.....	103
Gambar 4.43 Kurva <i>Throughput Network Coding Indoor</i> pada Modulasi M-QAM.....	105
Gambar 4.44 Kurva <i>Throughput Network Coding Outdoor</i> pada Modulasi M-QAM.....	107

## DAFTAR TABEL

Tabel 3.1 Parameter Sistem Implementasi WARP.....	39
---	----





# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Teknologi telekomunikasi menjadi kebutuhan yang telah melekat dan tak terelakan lagi untuk setiap individunya di dunia yang maju saat ini. Sistem komunikasi nirkabel yang merupakan bagian dari teknologi telekomunikasi kini menjadi solusi bagi setiap individu yang memiliki mobilitas yang cukup tinggi. Namun tidak sampai disitu, tingkat jumlah orang yang menginginkan mobilitas cepat, kebutuhan akan akses data cepat, dan keamanan akses data yang terus meningkat pula.

Telah diperkenalkan model baru yang dikenal dengan nama “*Network Information Flow*” yang terinspirasi dari aplikasi jaringan komputer. Sebuah komunikasi *point to point* yang menjadi dasar dengan sejumlah sumber informasi yang di-*multicast* ke tujuan yang telah ditentukan. Pada [1] dihasilkan sebuah karakteristik *coding rate* sederhana yang dapat diterima. Model ini tidak dapat diibaratkan sebagai “cairan” yang dapat dengan mudah diarahkan dan dibuat ulang. Namun dengan menerapkan *coding* pada *node*, yang dapat disebut sebagai *Network Coding*, *bandwidth* secara utuh dapat terselamatkan.

Pada bidang telekomunikasi khususnya sistem komunikasi nirkabel, informasi yang dikirim ke tujuan tertentu harus melalui tahap modulasi terlebih dahulu. Modulasi adalah suatu teknik penumpangan frekuensi sinyal informasi terhadap frekuensi pembawa (*carrier*) yang mempunyai frekuensi lebih tinggi dari sinyal informasi dan sesuai dengan media transmisi tempat sinyal akan dikirimkan. Jenis-jenis modulasi dibedakan sesuai dengan jenis sinyal informasi (analog dan digital) yaitu, Modulasi Analog (*Amplitude Modulation*, *Frequency Modulation*, dan *Phase modulation*), Modulasi Pulsa (*Pulse Amplitude Modulation*, *Pulse Position Modulation*, *Pulse Width Modulation*, dan *Pulse Code Modulation*), dan Modulasi Digital (*Frequency Shift Keying*, *Binary Pulse Shift Keying*, *Quadrature Phase Shift Keying*, dan *Quadrature Amplitude Modulation*) [2].

Bagaimana jika informasi yang dibawa oleh *network coding* untuk dikirim ke tujuan dimodulasi dengan modulasi *Quadrature Amplitude Modulation* (QAM). Modulasi QAM merupakan salah

satu dari modulasi digital yang mempunyai skema pensinyalan multisimbol. Pensinyalan tersebut dibangkitkan dengan dua pembawa dalam kuadratur fasa satu sama lain ( $\cos \omega_c t$  dan  $\sin \omega_c t$ ) ditransmisikan serentak melalui saluran yang sama [2]. Muncul sebuah konsep *network coding* pada sistem komunikasi nirkabel modulasi QAM. Konsep tersebut diimplementasikan menggunakan WARP. Platform WARP dikategorikan sebagai *platform* yang dapat mengimplementasikan algoritma *wireless* yang sangat kompleks [3]. Dari implementasi tersebut dilakukan pengukuran dan dilakukan analisis dari kinerja sistem tersebut.

## 1.2 Perumusan Masalah

- Sebuah konsep sistem komunikasi nirkabel modulasi QAM dengan *Network Coding* dapat diterapkan dan disimulasikan atau tidak.
- Konsep sistem komunikasi nirkabel modulasi QAM dengan *Network Coding* dapat diimplementasikan ke modul WARP atau tidak.
- Lebih baik atau tidak sistem komunikasi nirkabel modulasi QAM dengan *Network Coding* dan tanpa *Network Coding*.

## 1.3 Batasan Masalah

Permasalahan-permasalahan yang ada dalam prakteknya akan diselesaikan dengan batasan-batasan sebagai berikut:

- Matlab yang digunakan merupakan matlab versi R2013b.
- Implementasi menggunakan WARP dengan komponen utama: WARP FPGA Board v2.2 dan WARP Radio Board v1.4.
- *Framework* yang digunakan yaitu *framework* WARPLab 7.
- Modulasi yang digunakan yaitu modulasi M-QAM.
- *Band* frekuensi *carrier* yang digunakan yaitu 2.4 GHz.
- Pengukuran dilakukan *indoor* maupun *outdoor* dengan kondisi perubahan *gain* pada pengirim dan perubahan jarak.

## 1.4 Tujuan Penelitian

Hasil yang akan dicapai dari permasalahan yang diteliti bertujuan untuk:

- Menghasilkan konsep komunikasi *relay* modulasi M-QAM dengan *Network Coding*.

- Membuktikan bahwa *Network Coding* dapat diaplikasikan pada Layer Aplikasi.
- Menerapkan modulasi M-QAM dengan *Network Coding* menggunakan *Wireless Open-Access Research Platform* (WARP).
- Membuktikan performasi kinerja sistem komunikasi dengan *Network Coding* lebih baik dari sistem komunikasi tanpa *Network Coding*.

## **1.5 Metodologi Penelitian**

Tugas akhir ini dikerjakan dengan menggunakan metodologi sebagai berikut:

### **1.5.1 Studi Literatur**

Studi literatur tentang tema-tema terkait Tugas Akhir. Tema-tema terkait tugas akhir tersebut yaitu:

- Teori *Quadrature Amplitude Modulation* (QAM)
- Teori *Network Coding*
- Simulasi menggunakan *Software Matlab*
- Implementasi ke modul *Wireless Open-Access Research Platform* (WARP)

### **1.5.2 Pembuatan Simulasi Menggunakan *Software Matlab***

Studi literatur mengenai teori QAM dan *Network Coding* dibuktikan dengan melakukan simulasi. Simulasi dilakukan dengan menggunakan *Software Matlab*. Dari teori-teori yang telah didapatkan, maka akan dicoba mensimulasikan *Network Coding* dengan modulasi QAM. Hasil simulasi tersebut akan memudahkan dalam pemrograman ke modul WARP. Sehingga hanya butuh penyesuaian saja terhadap implementasi ke modul WARP. Hasil simulasi juga sebagai indikasi bahwa *syntax* yang telah dibuat bekerja sesuai dengan teori dan siap diimplementasikan ke modul WARP.

### **1.5.3 Implementasi ke Modul WARP**

Implementasi ke modul WARP ini menerapkan hasil simulasi ke alat untuk mendapatkan hasil nyata diluar dari hasil simulasi yang mempunyai kondisi serba ideal. Implementasi dilakukan dengan melakukan pemrograman WARPLab dan Matlab.



Pemrograman Matlab sebagian besar telah dikerjakan di simulasi. Sehingga tidak memerlukan proses yang panjang untuk dapat menyesuaikan dengan implementasi WARP.

#### **1.5.4 Uji Coba dan Perbaikan**

Uji coba dan perbaikan dilakukan terhadap dua metodologi lainnya, yaitu Simulasi Menggunakan *Software* Matlab dan Implementasi ke Modul WARP. Simulasi dilakukan uji coba, untuk uji coba modulasi QAM sendiri dan untuk uji coba *Network Coding* sendiri, kemudian ditinjau kesesuaiannya dengan teori yang ada. Hasil simulasi yang tidak sesuai dengan teori yang ada selanjutnya dilakukan evaluasi dan perbaikan.

#### **1.5.5 Pengukuran**

Setelah uji coba dan perbaikan selesai, dilakukan pengukuran. Parameter yang diukur yaitu sistem dari *Network Coding* dengan kondisi yang berbeda-beda. Kondisi yang dimaksud yaitu kondisi *Indoor* dan *Outdoor* dengan jarak yang berbeda-beda.

#### **1.5.6 Analisis**

Simulasi *Network Coding* dengan modulasi QAM yang telah berhasil diimplementasikan ke modul WARP dilakukan pengukuran dan pengambilan data. Hasil pengukuran dianalisis dan diambil kesimpulan.

### **1.6 Sistematika**

Proses penelitian yang tersusun dalam laporan tugas akhir dapat dijabarkan sebagai berikut:

- BAB 1, Bagian awal pada laporan tugas akhir akan membahas tentang latar belakang, tujuan dan permasalahan-permasalahan yang dihadapi dalam pengerjaan tugas akhir.
- BAB 2, Bagian ini membahas mengenai teori-teori terkait yang membantu dalam proses perumusan masalah sistem. Teori-teori tersebut juga digunakan sebagai acuan awal dalam penyelesaian permasalahan.
- BAB 3, Bagian ini membahas tentang langkah-langkah mendesain Sistem Komunikasi dengan modulasi M-QAM dan mengimplementasikannya ke modul WARP. Tahapan ini

bertujuan untuk menjelaskan proses desain sistem simulasi dan desain dalam bentuk implementasi pada WARP.

- BAB 4, Bagian ini akan membahas mengenai sistematika pengukuran dan hasilnya. Pada bab tersebut juga dilakukan analisis awal terhadap data-data hasil pengukuran.
- BAB 5, Bagian ini akan membahas kesimpulan berdasarkan hasil-hasil analisis pada bab sebelumnya.

### **1.7 Manfaat**

Tercapainya hasil dari permasalahan yang diteliti ini diharapkan dapat memberikan manfaat sebagai berikut:

- Memberikan kontribusi dalam mengembangkan penerapan *Network Coding* pada Layer Aplikasi.
- Menimbulkan aspirasi dalam meneliti dari permasalahan selanjutnya.





## BAB II

### TEORI PENUNJANG

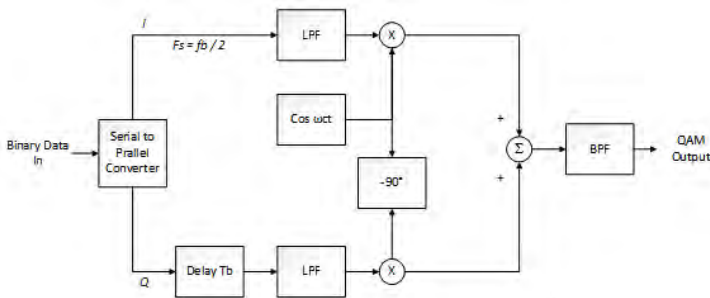
Pada bab ini berisikan dasar-dasar teori yang membantu dalam proses perumusan masalah sistem dan menjadi penunjang gagasan pemikiran awal sepanjang pelaksanaan Tugas Akhir, dari perancangan sistem, implementasi sistem, persiapan alat implementasi, hingga analisis data. Dasar-dasar teori tersebut dijabarkan sebagai berikut:

#### 2.1 *Quadrature Amplitude Modulation (QAM)*

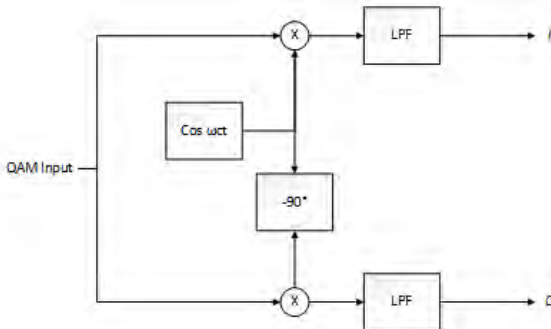
Pada komunikasi digital, transmisi data digital meningkat pada laju yang sangat cepat dan sinyal-sinyal analog sering di konversikan ke dalam format digital sebelum ditransmisikan. Saluran ini berupa medium yang memisahkan antara pentransmisi dengan penerima dapat berupa udara, kawat, tabung pengonduksi kosong (*waveguide*), atau sekelompok serat optik. Transmisi sinyal-sinyal pada medium perlu diefisiensikan dengan menumpangkan sinyal-sinyal orisinil (*baseband*) yang mengandung informasi ke sinyal-sinyal pembawa (*carrier*). Efisiensi transmisi mensyaratkan bahwa sinyal-sinyal pembawa informasi diolah dalam suatu cara sebelum ditransmisi melalui medium yang terletak antara pentransmisi dan penerima [2].

Umumnya, agar transmisi efisien sinyal-sinyal pita dasar (*baseband*) harus digeser ke frekuensi-frekuensi yang lebih tinggi. Hal ini dilakukan dengan mengubah-ubah amplitude, fasa, atau frekuensi. Proses tersebut dinamakan modulasi [2].

Pada sistem modulasi digital biner dalam pertimbangannya, hanya satu dari dua kemungkinan sinyal dapat terkirim sepanjang jarak tiap sinyal. Sistem modulasi digital mempunyai teori efisiensi *bandwidth* 1 bps/Hz. Pada beberapa aplikasi, sebuah sistem transmisi lebih hemat jika, dalam sebuah *bandwidth* tertentu, lebih banyak lagi bit per detik yang dapat ditransmisikan. Hal itu membawa ke pertimbangan *M*-ary metode modulasi digital yang mana satu dari *M* kemungkinan sinyal ditransmisikan sepanjang jarak sinyal. Pendekatan awal untuk meningkatkan efisiensi spektrum yaitu dengan menggunakan prinsip *Quadrature Multiplexing* dimana 2 sinyal termodulasi digabung pada kuadratur fasa. Berikut merupakan sistem dasar yang disebut *Quadrature Amplitude Modulation (QAM)* [4].



**Gambar 2.1** Blok Diagram Pemancar QAM [4]



**Gambar 2.2** Blok Diagram Penerima QAM [4]

Pada sistem QAM diatas, *serial to parallel converter* urutan data biner  $f_b = 1/T_b$  bps, dan diteruskan secara paralel  $f_s = 1/T_s$  bps, dimana  $T_s = 2T_b$ . Di rumus tersebut  $f_s$  adalah *symbol rate* atau *baud rate*, dan  $f_b$  adalah keseluruhan dari sistem *bit rate*. Kemudian *low pass filter* digunakan untuk membatasi *bandwidth* dan menyediakan pembentukan spektrum. Sinyal data I dan Q dimodulasi menggunakan modulasi DSB-SC; sinyal I menggunakan referensi *phase carrier*; dan sinyal Q menggunakan referensi *quadrature carrier*. modulasi sinyal I dan Q dijumlahkan menjadi sebuah sinyal QAM [4].

### 2.1.1 Penentuan Amplitudo Sinyal

Pada proses modulasi M-QAM, aliran bit-bit pertama kali dipetakan dengan *Gray Mapping* (*Gray Coding*) ke dalam koordinat konstelasi sinyal biner *In-phase* (I) dan *Quadrature* (Q)

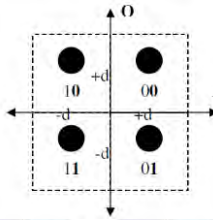
seperti Gambar 2.1, sedangkan sebaliknya pada proses demodulasi M-QAM, aliran sinyal-sinyal analog akhirnya dikembalikan menjadi sinyal-sinyal biner dengan menggunakan *Gray Mapping* seperti Gambar 2.2. M adalah jumlah multilevel modulasi atau jumlah maksimum data pada M-QAM dalam bentuk bit. Untuk sistem modulasi 4-QAM dapat dikirimkan 4 data dalam biner, sedangkan untuk 64 QAM dapat dikirimkan 64 data dalam biner. Jumlah untaian biner yang dapat mempresentasikan data disebut dengan ukuran konstelasi bit, disimbolkan dengan k, dan dapat dinyatakan sebagai [5]:

$$k = \log_2 M \quad (2.1)$$

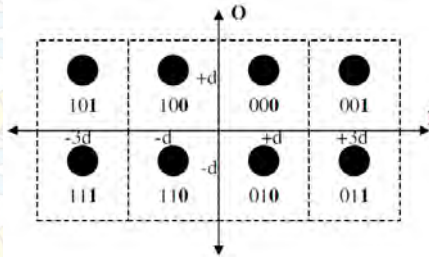
### 2.1.2 *Gray Mapping (Gray Coding)* [6]

Pemetaan sinyal pada modulasi M-QAM dengan *Gray Mapping* (Gray Coding) dapat dilakukan dengan mengkombinasikan sisi tegak *Quadrature* (Q) dan sisi sudut *In-phase* (I) konstelasi sinyal. Angka-angka yang secara umum digunakan untuk menyatakan nilai AMq sebagai sisi tegak Q dan nilai AMi sebagai sisi sudut I adalah konstanta d. Untuk modulasi 4-QAM, terdapat pasangan (d, -d) dimana nilai d menyatakan bit 0 dan -d menyatakan bit 1. Untuk 8-QAM pada sisi sudut I ditambahkan nilai (-3d, +3d), dimana pada penambahan titik -3d dan +3d, ditambahkan bit 1, sedangkan pada -d dan +d, ditambahkan bit 0. Untuk 16-QAM pada sisi tegak Q dan sisi sudut I ditambahkan nilai (-3d, +3d), dimana pada penambahan titik -3d dan +3d di sisi tegak Q ditambahkan bit 1, sedangkan pada -d dan +d ditambahkan bit 0. Untuk 32-QAM, pada sisi sudut I ditambahkan nilai (-5d, +5d) dan (-7d, +7d), sedangkan pada 64-QAM pada sisi sudut I dan sisi tegak Q ditambahkan nilai (-5d, +5d) dan (-7d, +7d), seperti ditunjukkan pada Gambar 2.3, Gambar 2.4, Gambar 2.5, Gambar 2.6, dan Gambar 2.7 sebagai berikut.

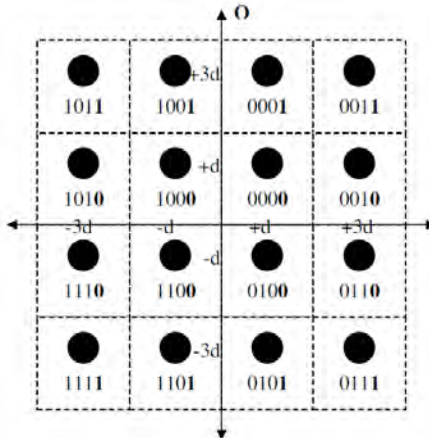




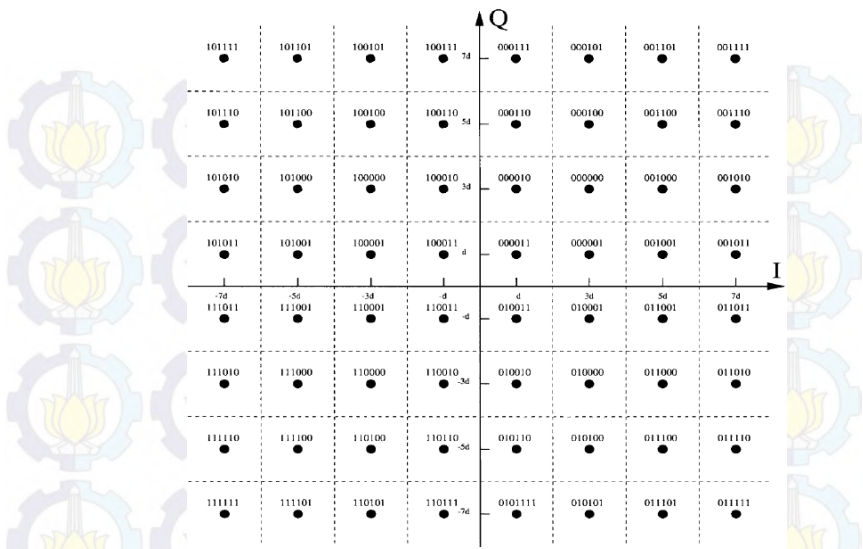
**Gambar 2.3** Konstelasi Sinyal 4-QAM



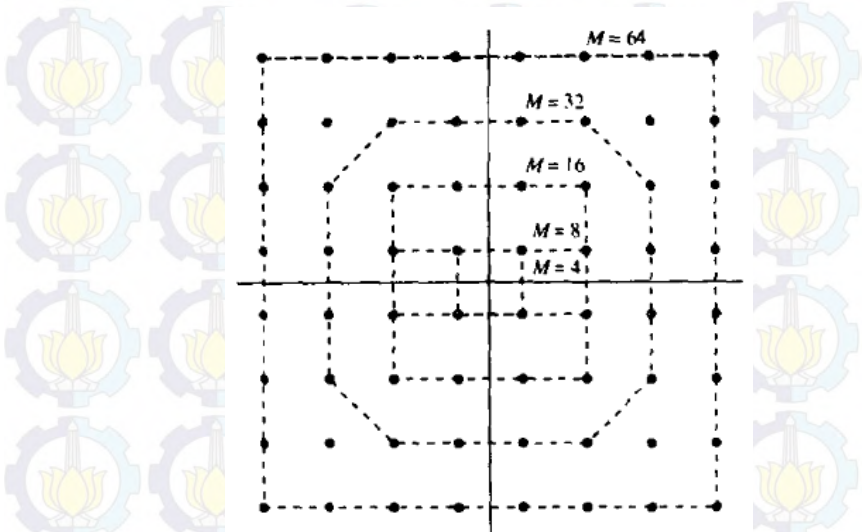
**Gambar 2.4** Konstelasi Sinyal 8-QAM



**Gambar 2.5** Kontelasi Sinyal 16-QAM



**Gambar 2.6** Konstelasi Sinyal 64-QAM



**Gambar 2.7** Diagram Ruang Sinyal untuk Rectangular QAM

### 2.1.3 Probabilitas *Bit Error* pada M-QAM [6]

Pada M-QAM sinyal hasil modulasi M-QAM dapat dinyatakan ke dalam persamaan

$$S_m(t) = A_{mc} g(t) \cos 2\pi f_c t - A_{ms} g(t) \sin 2\pi f_c t$$

$$0 \leq t \leq T \quad (2.2)$$

Dimana  $A_{mc}$  dan  $A_{ms}$  adalah amplitudo sinyal dari sinyal pembawa quadrature dan  $g(t)$  adalah pulsa sinyal. Untuk menentukan probabilitas kesalahan (*error*) pada QAM, kita harus menetapkan sinyal ke dalam titik konstelasi. Pertama melihat sinyal 4-QAM dengan  $M = 4$ . Gambar 2.8 menggambarkan dua konstelasi 4-QAM. Gambar 2.8 (a) adalah empat fasa sinyal modulasi dan Gambar 2.8 (b) adalah sinyal 4-QAM dengan dua level amplitudo yaitu  $A_1$  dan  $A_2$ , dan empat fasa. Karena probabilitas eror didominasi oleh jarak minimum antara titik pada konstelasi, maka dapat menentukan kondisi dengan  $d_{min}(e) = 2A$  untuk konstelasi sinyal. Dan juga menentukan nilai daya transmisi rata-rata, dengan alasan semua titik sinyal memungkinkan. Untuk 4 fasa sinyal, digunakan persamaan 2.3.

$$P_{zv} = \frac{1}{4} (4) 2A^2 = 2A^2 \quad (2.3)$$

Untuk dua amplitudo, empat fasa 4-QAM diletakkan titik-titik pada lingkaran. Dimana  $d_{min}(e) = 2A$  sehingga daya transmisi rata-rata sinyal adalah

$$P_{zv} = \frac{1}{4} [2(3A^2) + 2A^2] = 2A^2 \quad (2.4)$$

Dimana konstelasi sinyal untuk 4-QAM fasa, memiliki *error rate* yang sama untuk kedua sinyal pada Gambar 2.8.

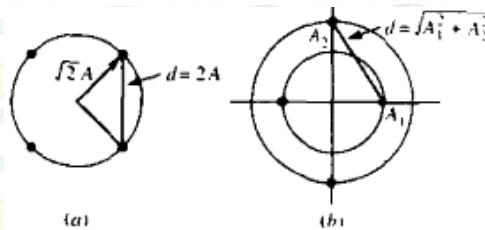
Dengan kata lain tidak ada keuntungan dari dua amplitudo sinyal 4-QAM untuk fase modulasi. Selanjutnya untuk  $M = 8$ QAM, pada kasus ini banyak kemungkinan konstelasi sinyal, dapat dilihat pada Gambar 2.9. dari empat gambar pada Gambar 2.9 mempunyai dua amplitudo dan memiliki jarak titik minimal dari  $2A$ . Dengan koordinat ( $A_{mc}$ ,  $A_{ms}$ ) untuk beberapa titik sinyal dinormalisasi  $A$ . Diasumsikan titik sinyal sama-sama mungkin, maka daya transmisi rata-rata sinyal dapat ditentukan dengan menggunakan persamaan (2.5)

$$P_{zv} = \frac{1}{M} \sum_{m=1}^M (A_{mc}^2 + A_{ms}^2)$$

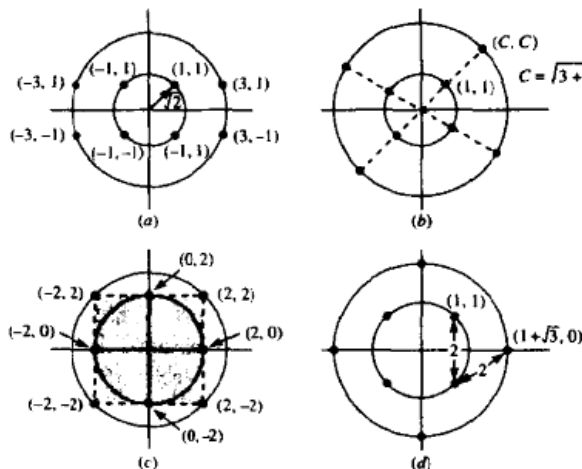
$$P_{zv} = \frac{A^2}{M} \sum_{m=1}^M (a_{mc}^2 + a_{ms}^2) \quad (2.5)$$



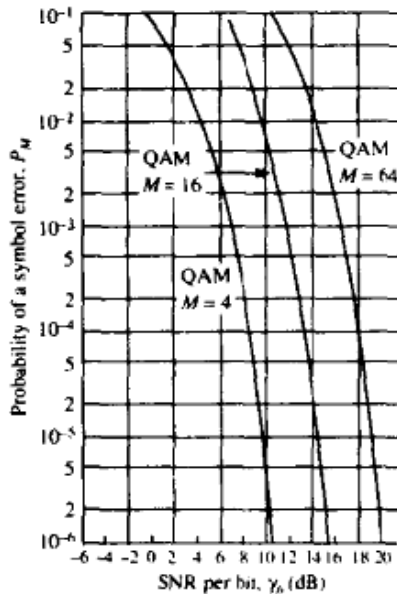
Dimana  $(A_{mc}, A_{ms})$  adalah koordinat dari titik sinyal yang dinormalisasi A. Untuk Gambar 2.9 (a) dan (c) memiliki titik dengan bentuk rectangular dan memiliki  $P_{av} = 6A^2$ . Dan untuk Gambar 2.9 (b) memiliki rata-rata daya transmisi sebesar  $P_{av} = 6.38 A^2$  dan (d)  $P_{av} = 4.73 A^2$ . Oleh karena itu sinyal (d) pada Gambar 2.9 membutuhkan daya kira-kira 1 dB lebih kecil dibandingkan (a) dan (b) dan 1,6 lebih sedikit dibandingkan (c) untuk mencapai probabilitas error yang sama. Untuk grafik perbandingan probabilitas error untuk M-QAM dapat dilihat pada Gambar 2.10



**Gambar 2.8** Dua Konstelasi Sinyal QAM dengan  $M = 4$



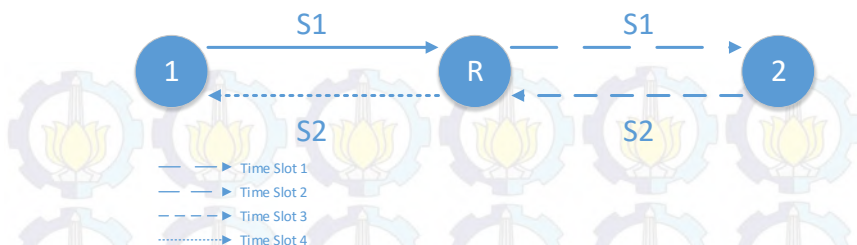
**Gambar 2.9** Empat Konstelasi Sinyal untuk 8-QAM



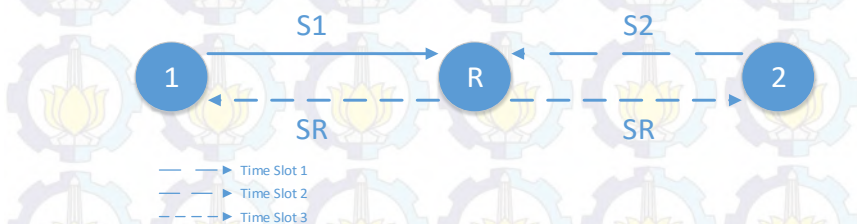
**Gambar 2.10** Grafik Probabilitas Error Simbol M-QAM

## 2.2 Network Coding

Pada jaringan computer yang ada, tiap *node* berfungsi sebagai sebuah *switch* dalam pengertian meneruskan salah satu informasi dari sebuah *input link* ke sebuah *output link*, atau membuat salinan informasi yang diterima dari sebuah *input link* dan mengirimnya ke *output link* yang telah ditentukan. Dari sudut pandang *Information Theoretic* (Teori Informasi), tidak ada alasan untuk membatasi fungsi dari sebuah *node* yang menjadi *switch*. Lebih baik, sebuah *node* dapat berfungsi sebagai sebuah *encoder* dalam pengertian *node* menerima informasi dari semua *input link*, meng-*encode*, dan mengirim informasi ke semua *output link*. Dari sudut pandang ini, sebuah *switch* merupakan sebuah *encoder* spesial. Selanjutnya, itu semua akan mengarah ke *coding* pada sebuah *node* pada sebuah jaringan sebagai *Network Coding* [1].



**Gambar 2.11** Skema *Tradisional Non-Network Coding* [7]



**Gambar 2.12** Skema *Straightforward Network Coding* [7]

### 2.2.1 Skema *Non-Network Coding* [7]

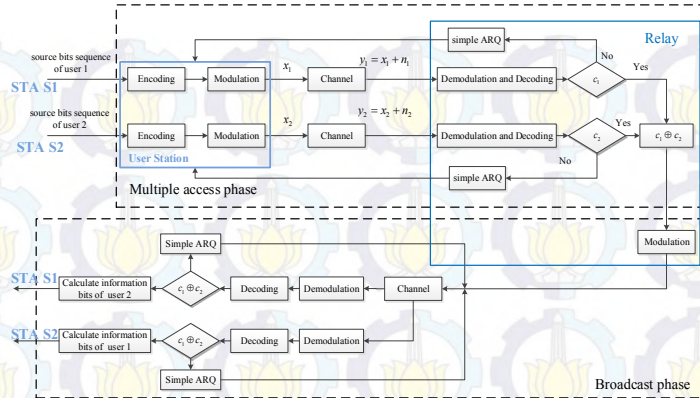
Tanpa penggunaan *network coding*, dan dengan konfigurasi dasar yang mencoba untuk menghindari interferensi, sejumlah empat *timeslot* dibutuhkan untuk bertukar dua paket, satu pada setiap tujuan. Gambar 2.11 menggambarkan proses pertukaran dua paket. Pada *timeslot* 1, *node* 1 mengirimkan sebuah paket  $S_1$  ke *relay* R. pada *timeslot* 2, *relay* R meneruskan paket  $S_1$  ke *node* 2. Pada *timeslot* 3, *node* 2 mengirimkan sebuah paket  $S_2$  ke *relay* R. Dan pada *timeslot* 4, *relay* R meneruskan paket  $S_2$  ke *node* 1.

### 2.2.2 Skema *Straightforward Network Coding*

Skema *straightforward* dari aplikasi *network coding* dapat mengurangi jumlah *timeslot* menjadi tiga *timeslot* dari jumlah sebelumnya yaitu empat *timeslot*. Skema ini menghasilkan perbaikan *throughput* sebesar 33% dari skema *non-network coding* [7].

Pada Gambar 2.12 menggambarkan sebuah skema dari *straightforward network coding*. Pada *timeslot* 1, *node* 1 mengirimkan sebuah paket  $S_1$  ke *relay* R. Pada *timeslot* 2, *node* 2





**Gambar 2.13** Diagram Proses Skema *Straightforward Network Coding* [8]

mengirimkan sebuah paket  $S_2$  ke *relay* R. Setelah menerima paket  $S_1$  dan  $S_2$ , *relay* R membuat sebuah paket  $S_r = f(S_1, S_2)$ . Pada umumnya, beberapa pemetaan *network coding*  $f(S_1, S_2)$  adalah mungkin. Formula  $f(S_1, S_2)$  adalah operasi XOR, dinyatakan sebagai berikut:

$$S_r = S_1 \oplus S_2 \quad (2.6)$$

dimana  $\oplus$  menyatakan pasangan XOR simbol per simbol dari paket  $S_1$  dan  $S_2$ . Pada *timeslot* ke 3, *relay* R mem-broadcast  $S_r$  ke *node* 1 dan *node* 2. Pada saat *node* 1 menerima informasi dari  $S_r$ , mengekstrak informasi  $S_2$  dari  $S_r$  menggunakan informasi dari  $S_1$ . Sehingga dapat ditulis sebagai berikut:

$$S_1 \oplus S_r = S_1 \oplus (S_1 \oplus S_2) = S_2 \quad (2.7)$$

demikian juga pada *node* 2, mengekstrak informasi  $S_1$  dari  $S_2 \oplus S_r$  [7]. Diagram proses Skema *Straightforward Network Coding* digambarkan pada Gambar 2.13.

### 2.2.3 Pertimbangan Noise dan Throughput [7]

*Noise* menjadi suatu kehadiran mendasar untuk semua sistem komunikasi. Selama penelitian, *noise* diabaikan. Analisis *throughput* didasarkan pada waktu perhitungan slot, dengan asumsi apapun yang dikirim akan diterima dengan benar. Sebuah

analisis lebih mendalam harus mencakup pertimbangan *noise*. Dengan *noise* dinyatakan

$$y_R = a_1 + jb_1 + a_2 + jb_2 + w_R \quad (2.8)$$

dimana  $w_R$  merupakan *noise* biasanya dimodelkan sebagai sebuah *Gaussian Random Variable*.

Dengan *noise*, apakah *bit error rate* (BER) akan relative meningkat terhadap *non-network coding* atau *straightforward network coding*. Ternyata dengan modulasi QPSK, *end-to-end* BER diantara *node* 1 dan *node* 2 dari *Physical Network Coding* sebanding dengan *non-network coding*, dan sebenarnya sedikit lebih baik dari *straightforward network coding*, seperti yang dijelaskan pada tiga paragraph berikutnya.

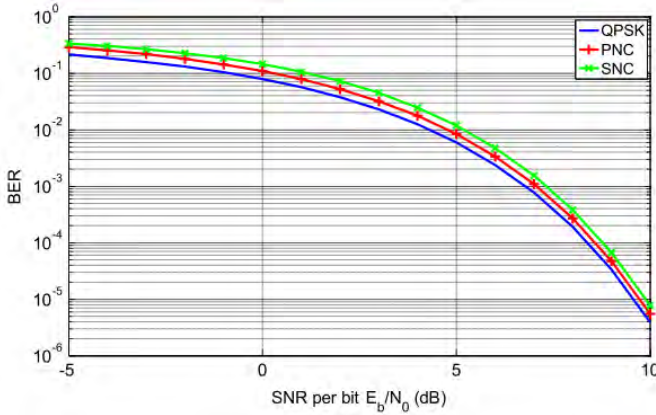
Asumsikan  $P_e$  sebagai BER dari QPSK klasik saluran *point to point*. Dengan demikian, *non-network coding* terdiri dari empat layaknya satu *hop*. Pada *non-network coding*, *straightforward network coding*, dan *physical network coding* dibahas di bawah, agar sederhana daya pancar diasumsikan sama untuk semua *node*, *channel gain* di semua arah yang sama, dan daya terima *noise* yang sama di semua *node*. Kemudian  $P_e$  sama di semua saluran satu *hop* pada *non-network coding*. *End-to-end* BER dari *non-network coding* adalah

$$2(1 - P_e)P_e \quad (2.9)$$

Itu merupakan sebuah bit terkirim dalam kesalahan di bawah sua situasi: (i) ada kesalahan deteksi di *hop* pertama, tapi tidak ada kesalahan deteksi di *hop* kedua, atau (ii) tidak ada kesalahan deteksi di *hop* pertama, tetapi ada kesalahan deteksi di *hop* kedua.

BER dari saluran *point to point* QPSK klasik,  $P_e$  diplot pada Gambar 2.14. Sebaai perbandingan BER dari  $a_r$  (or  $b_r$ ) pada *uplink* XOR sistem *physical network coding* yang juga diplot pada Gambar 2.14. Untuk *physical network coding* dan *straightforward network coding* target sinyal pada *relay* adalah sinyal XOR. Dengan demikian BER adalah BER XOR *decoding*, bukan BER dari bit individu dari kedua tujuan akhir. Tercatat bahwa pada Gambar 2.14 *uplink* BER dari  $a_r$  (or  $b_r$ ) pada *physical network coding* kira-kira sama dengan BER dari saluran QPSK *point to point*,  $P_e$ . Oleh karena itu, sama seperti argumen pada paragraf sebelumnya, BER *end-to-end* dari *physical network coding* adalah sekitar

$$2(1 - P_e)P_e \quad (2.10)$$



**Gambar 2.14** Kurva BER QPSK, PNC, dan SNC

sama seperti di *non-network coding*. Sehingga bagaimanapun *physical network coding* menggunakan dua *timeslot* dan *non-network coding* menggunakan empat *timeslot*. Dengan demikian, pada BER yang sama, *throughput physical network coding* dua kali lipat dari *non-network coding*.

Untuk BER *straightforward network coding* dari  $a_r$  (or  $b_r$ ) pada *relay* adalah

$$2(1 - P_e)P_e \quad (2.11)$$

Ada sebuah *error XOR decoding* jika bit dari *node 1* di-decode dengan *error* dan bit dari *node 2* di-decode benar dan sebaliknya. Dengan demikian, BER *uplink* dari SNR (yang diplot pada Gambar 2.14) adalah sama dengan BER *end-to-end* dari *non-network coding* dan *physical network coding*. BER *end-to-end* dari *straightforward network coding* adalah

$$\begin{aligned} & 2(1 - P_e)P_e \cdot (1 - P_e) + [1 - 2(1 - P_e)P_e] \cdot P_e = \\ & 3P_e - 6P_e^2 + 4P_e^3 > 2(1 - P_e)P_e \end{aligned} \quad (2.12)$$

untuk  $P_e > 0$ . *Straightforward network coding* tidak hanya mempunyai *throughput* lebih kecil dari *physical network coding*, tapi juga mempunyai BER yang lebih tinggi.

Untuk *non-network coding*, empat *hop* dibutuhkan untuk mengirim paket. Asumsi, waktu yang dibutuhkan untuk tiap *hop* dengan ARQ adalah sebuah distribusi geometri dengan rata-rata



$1/(1 - P_e)$ . Demikian, empat *hop* membutuhkan  $4/(1 - P_e)$  waktu unit diatas rata-rata. *Throughput* adalah

$$Th_{TS} = (1 - P_e)/2 \quad (2.13)$$

Untuk *straightforward network coding*, dua *uplink hop* mengkonsumsi  $2/(1 - P_e)$  *timeslot* diatas rata-rata. *Downlink hop* mem-broadcast informasi yang sama untuk dua *node* tujuan. Setelah transmisi pertama, mungkin ada diantara *node* tujuan sukses menerima paket, hanya satu dari *node-node* tersebut yang menerima itu dengan sukses, atau tidak ada dari *node-node* tersebut yang menerima. Asumsi  $T_0$  merupakan waktu rata-rata yang dibutuhkan untuk *downlink hop*. Asumsikan  $T_1$  merupakan waktu tersisa yang dibutuhkan untuk *downlink hop* setelah satu dari *node-node* telah menerima paket dengan sukses. Dengan pembaharuan argumen,

$$T_0 = 1 + P_e^2 T_0 + 2P_e(1 - P_e)T_1 \text{ dan } T_1 = 1 + P_e T_1 \quad (2.14)$$

Pemecahan rumus menghasilkan

$$T_0 = (1 + 2P_e)/(1 - P_e^2) \quad (2.15)$$

Dengan demikian,

$$Th_{SNC} = \frac{2}{\frac{2}{1-P_e} + T_0} = 2(1 - P_e^2)/(3 + 4P_e) \quad (2.16)$$

Untuk *physical network coding*, waktu *uplink* adalah  $1/(1 - P_e)$  pada rata-rata. Waktu *downlink* adalah  $T_0$  pada rata-rata, pada *straightforward network coding*. Dengan demikian,

$$Th_{PNC} = \frac{2}{\frac{1}{1-P_e} + T_0} = 2(1 - P_e^2)/(2 + 3P_e) \quad (2.17)$$

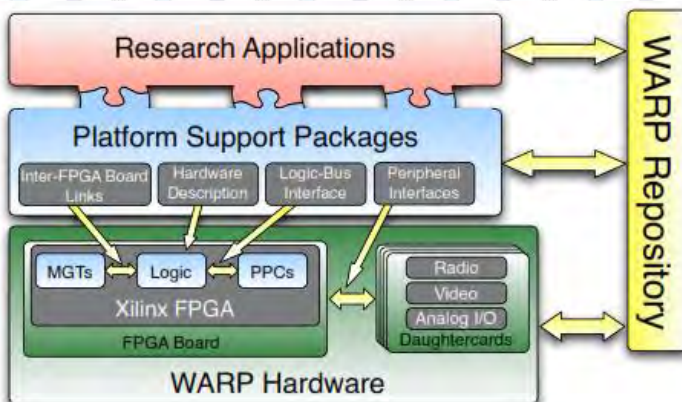
## 2.3 Wireless Open-Access Research Platform (WARP)

*Wireless Open-Access Research Platform* (WARP) merupakan sebuah desain platform yang mampu, terukur, dan dapat diperluas dengan sangat baik untuk penelitian nirkabel canggih. Tujuan dari platform ini dapat dijelaskan pada sebuah level tinggi melalui 4 kunci persyaratan. Pertama, platform ini mampu menerapkan algoritma nirkabel canggih. Kedua, platform harus terukur, memungkinkan tambahan sumber pengolahan untuk dialokasikan ketika daya komputasi dari sebuah pengolah tunggal membuktikan tidak mampu. Ketiga, harus dapat diperluas, memberikan pilihan perluasan Interface dan Peripheral untuk mengakomodasi aplikasi masa depan. Keempat, platform harus mendukung evolusi beberapa komponen tanpa mengorbankan fungsi lain [3].

### 2.3.1 Arsitektur Platform [3]

*Platform* ini didesain untuk memenuhi riset sistem komunikasi nirkabel dengan kinerja yang tinggi. Arsitektur *platform* ini terdiri dari empat komponen. Untuk lebih jelasnya bisa dilihat pada Gambar 2.15.

- *Custom Hardware*: Perangkat keras disesuaikan untuk memenuhi sistem komunikasi dengan kinerja yang tinggi. Perangkat keras ini menyediakan sumber daya untuk DSP berkecepatan tinggi yang bisa digunakan untuk implementasi algoritma yang intensif, terukur interkoneksi untuk mengalokasikan lebih banyak daya pemrosesan yang diperlukan, prosesor khusus, dan ekspansi *interface* yang lainnya.
- *Platform Support Packages*: Alat desain dan modul antarmuka tingkat rendah dirancang untuk memungkinkan penggunaan *hardware* oleh para peneliti di seluruh lapisan desain jaringan nirkabel.
- *Open-Access Repository*: Pusat penyimpanan untuk semua *source code*, model dan *file* desain *hardware*. *Content* secara lengkap akan tersedia di bawah lisensi BSD seperti *open-source*.



**Gambar 2.15** Komponen-komponen Platform

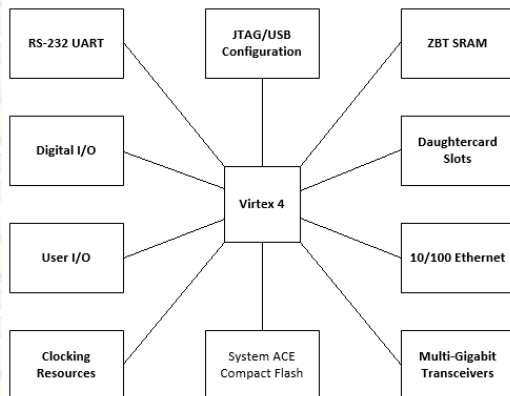
- *Research Applications*: Implementasi dari algoritma baru, diaktifkan oleh *hardware* khusus dan paket mendukung

*platform*. Sistem Kendali menggabungkan algoritma baru dapat dengan cepat dibangun menggunakan bangunan standar modul blok yang disediakan dalam *repository*.

Dalam desain, terdapat dua metode yang bisa digunakan untuk meneliti desain *physical layer* (PHY). Metode pertama adalah menggunakan sistem *non-real-time* yaitu dengan WARPLab. Sistem ini menggunakan kondisi kanal riil, namun dengan pemrosesan data secara *offline* melalui MATLAB. Metode kedua adalah membangun sistem *real-time* yang diinginkan untuk diimplementasikan pada FPGA (*Field Programmable Gate Array*). Sistem tersebut dapat dibangun secara independen dan berdiri sendiri pada *node* WARP.

### 2.3.2 WARP FPGA Board v2

Komponen utama *hardware* dari WARP adalah FPGA board, radio board, dan clock board. FPGA board terdiri dari FPGA chips, 4Mb *onboard memory* dan memiliki beberapa *pheriperall* I/O yaitu multi gigabit *transceiver*, *serial port*, *ethernet port*, *daughtercard connectors*, *compact flahs slot*, JTAG header dan USB port. Arsitektur hardware WARP dapat dilihat pada Gambar 2.16 dan bentuk fisik dari WARP FPGA v2 pada Gambar 2.17 [9].



**Gambar 2.16** Arsitektur *Hardware* WARP FPGA [10]





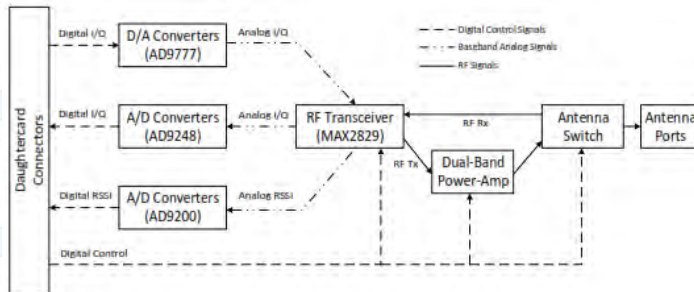
**Gambar 2.17** Bentuk Fisik WARP FPGA Board v2 [10]

### 2.3.3 WARP Radio Board v1.4

*Radio board* pada WARP didesain menggunakan IC Maxim MAX2829. *Single-chip* MAX2829 merupakan IC RF *transceiver* yang didesain khusus untuk aplikasi OFDM 802.11 WLAN yang dapat bekerja pada frekuensi RF 2.4 GHz-2.5 GHz dan 4.9 GHz-5.875 GHz. *Radio board* juga dilengkapi dengan *analog to digital* (ADC) dan *digital to analog* (DAC) *converter*. Untuk ADC *radio board* ini menggunakan IC AD9248 dan untuk DAC menggunakan IC AD9777 yang merupakan produk dari *analog device*. Bentuk fisik dari *radio board* dapat dilihat pada Gambar 2.18 dan arsitektur dari *radio board* WARP dapat dilihat pada Gambar 2.19 [9].



**Gambar 2.18** Bentuk Fisik WARP *Radio Board* v1.4 [11]



**Gambar 2.19** Arsitektur *Radio Board* v1.4 [11]

#### 2.3.4 WARPLab

WARPLab merupakan *framework* untuk *prototype physical layer* dengan MATLAB dan *hardware* WARP. Dengan WARPLab, kita dapat secara langsung membangkitkan sinyal dari MATLAB dan dapat ditransmisikan secara *realtime* melalui udara menggunakan node WARP. WARPLab memfasilitasi *prototyping* cepat dari algoritma *physical layer* langsung dari MATLAB. Arsitektur WARPLab *framework* ditunjukkan pada Gambar 2.20 [9].

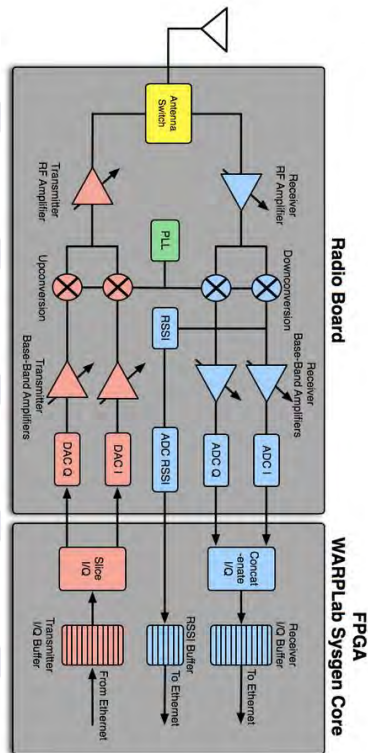


**Gambar 2.20** WARPLab Framework [9]

Referensi rancangan WARPLab adalah sebuah implementasi dari kerangka kerja WARPLab yang dapat melakukan satu atau lebih rancangan *physical layer* dapat dibuat dan diuji. Referensi rancangan gabungan MATLAB dan implementasi FPGA dari modul WARPLab. MATLAB digunakan untuk mengendalikan *nodes* dan proses sinyal. WARPLab 7 yang terakhir dirancang agar dapat bekerja pada *hardware* WARP v2 dan WARP v3, sedangkan untuk *hardware* WARP v1 menggunakan WARPLab 6. Kelebihan dari WARPLab 7 adalah [9]:

- Objek baru yang bisa berorientasi dengan *framework* m-code
- Paket “*sniffing*” *logic* (kompatibel dengan WARP v3 saja)
- Mendukung untuk jumbo *Ethernet* hingga 9014 byte, menghasilkan peningkatan kerja yang signifikan (kompatibel dengan WARP v3 saja)
- *Cross platform* (diuji pada Windows dan Mac OSX)
- *Default buffer* 215 (32k) *sample* (durasi 800 *micro second* per siklus Tx/Rx)
- Xilinx *tools* terbaru (ISE 14.4)





**Gambar 2.21** Desain Arsitektur FPGA WARPLab [11]

Tx dan Rx untuk 1 antena diperlihatkan pada Gambar 2.21, blok-blok pada bagian pemancar ditandai dengan warna jingga dan blok-blok pada bagian penerima ditandai oleh warna biru [11].

Blok pada jalur pemancar [11]:

- **Tx I/Q Buffer:** sampel *In-Phase* yang akan dikirim disimpan dalam Tx I Buffer dan sampel *Quadrature* yang akan dikirim disimpan dalam Tx Q Buffer.
- **DAC I/Q:** *Digital-to-Analog Converter* untuk sampel *In-Phase/Quadrature*.

- Tx BB Amplifier: *Transmitter Base Band Amplifier*. Ada dua amplifier, satu untuk sinyal *In-Phase* dan satu untuk sinyal *Quadrature*. Ini adalah *variabel Gain Amplifier* yang keduanya diatur dengan nilai penguatan yang sama, yang dapat dimasukkan oleh *user* dari *workspace* Matlab.
- Upconversion: Mengubah sinyal base band ke sinyal RF. Frekuensi *carrier* tergantung pada pengaturan PLL.
- Tx RF Amplifier: *Transmitter RF amplifier*, *Variabel Gain Amplifier*, nilai nilai penguatan yang dapat dimasukkan oleh *user* dari *workspace* Matlab.

Blok pada jalur penerima [11]:

- Rx RF Amplifier: *Receiver RF amplifier*, *variabel gain amplifier*, nilai penguatan yang dapat dimasukkan oleh *user* dari *workspace* Matlab.
- Downconversion: Mengubah sinyal RF ke basis sinyal *In-Phase/Quadrature* Band. Downconversion dari RF tergantung pada pengaturan PLL.
- Rx BB Amplifier: *Receiver Basis Band Amplifier*. Ada dua amplifier, satu untuk sinyal I dan satu untuk sinyal Q. Ini adalah *variabel Gain Amplifier* yang keduanya diatur dengan nilai penguatan yang sama, yang dapat dimasukkan oleh *user* dari *workspace* Matlab.
- Rx RF Amplifier: *Receiver RF amplifier*, *variabel gain amplifier*, nilai penguatan yang dapat dimasukkan oleh *user* dari *workspace* Matlab.
- Downconversion: Mengubah sinyal RF ke basis sinyal *In-Phase/Quadrature* Band. Downconversion dari RF tergantung pada pengaturan PLL.
- Rx BB Amplifier: *Receiver Basis Band Amplifier*. Ada dua amplifier, satu untuk sinyal I dan satu untuk sinyal Q. Ini adalah *variabel Gain Amplifier* yang keduanya diatur dengan nilai penguatan yang sama, yang dapat dimasukkan oleh *user* dari *workspace* Matlab.
- RSSI: *Received Signal Strength Indicator*. Blok ini mengukur RSSI tersebut.
- ADC I/Q: *Analog-to-Digital Converter* untuk sampel *In-Phase/Quadrature*.

- ADC RSSI: Analog-to-Digital Converter untuk pengukuran RSSI. Data RSSI tersedia di 1/4 laju data I/Q.
- Rx I/Q Buffer: sampel *In-Phase* yang diterima akan disimpan dalam Rx Buffer I dan sampel *Quadrature* yang diterima akan disimpan di Rx Q Buffer.
- RSSI Buffer: RSSI data disimpan dalam buffer ini.

Blok PLL: Pengaturan PLL menentukan frekuensi pembawa yang dapat diatur ke salah satu dari 14 kanal di Wi-Fi 2,4 GHz band atau 23 saluran di 5 GHz. Saluran pembawa dapat diatur langsung dari *workspace* Matlab.

CATATAN: *Receiver RF dan BB gain* dapat diatur menggunakan *Gain Manual Control* (MGC) atau *Automatic Gain Control* (AGC). Dalam mode MGC pengguna mengambil nilai-nilai dari penguatan dan menetapkan penguatan pada modul menggunakan fungsi yang tersedia dalam Referensi WARPLab M-Code. Dalam mode AGC papan dikonfigurasi sehingga secara otomatis menetapkan penguatan penerima berdasarkan pengukuran RSSI. *Framework* WARPLab memberikan contoh yang menggambarkan bagaimana menggunakan mode MGC dan mode AGC.

### 2.3.5 Spesifikasi WARPLab [10]

#### Tx/Rx I/Q Buffers

- a. Tx/Rx I/Q Buffers yang *independent*
- b. Setiap *buffer* dapat menyimpan maksimum  $2^{14}$  sampel
- c. Frekuensi sampling yang digunakan adalah 40 MHz

#### Syarat Sinyal yang dipancarkan:

- a. Amplitudo pada bagian real (*In-Phase Signal: I*) adalah  $[-1,1]$  dan amplitudo pada bagian imajiner (*Quadrature Signal: Q*) adalah  $[-1,1]$ . Sinyal I/Q pada pemancar terwakili dalam FPGA menggunakan 15-16 titik tetap.
- b. Frekuensi terendah: 30 KHz (*Radio Filter DC*)
- c. Frekuensi tertinggi: bergantung pada pengaturan frekuensi pada *Tx/Rx Low Pas Filter* (LPF). Secara default, Tx dan Rx LPF diatur oleh *nominal mode*. Adapun pengaturannya adalah sebagai berikut:  
LPF pada pemancar



- 1) Mode 0: Undefined
- 2) Mode 1: 12 MHz (*Nominal Mode*)
- 3) Mode 2: 18 MHz (Turbo Mode 1)
- 4) Mode 3: 24 MHz (Turbo Mode 2)

LPF pada penerima

- 1) Mode 0: 7.5 MHz
- 2) Mode 1: 9.5 MHz (*Nominal Mode*)
- 3) Mode 2: 14 MHz (Turbo Mode 1)
- 4) Mode 3: 18 MHz (Turbo Mode 2)

#### I/Q ADC/DAC

- a. Clock selalu pada 40 MHz
- b. Masukan sinyal DAC I/Q direpresentasikan menggunakan titik tetap 16\_15
- c. Masukan sinyal ADC I/Q direpresentasikan menggunakan titik tetap 14\_13

#### Tx/Rx Amplifiers

- a. Tx RF PA: 30 dB
- b. Tx BB: 0-3 dB
- c. Tx RF: 0-63 dB
- d. Rx BB: 0-31 dB
- e. Rx RF: 0-63 dB

## **2.4 Bit Error Rate (BER) dan Throughput**

Performansi dari suatu sistem dinilai dari nilai *Bit Error Rate* (BER) dan *Throughput*. Kedua parameter tersebut juga sering digunakan sebagai dasar analisis. Nilai BER dan nilai *throughput* saling mempengaruhi satu sama lain. Perhitungan nilai BER berdasarkan jumlah bit yang salah sedangkan perhitungan nilai *throughput* berdasarkan jumlah bit yang benar. Akan tetapi, meskipun nilai BER dan nilai *throughput* saling berhubungan, kedua parameter tersebut merupakan dua hal yang berbeda. Nilai BER merupakan nilai rasio sedangkan nilai *throughput* merupakan nilai per satuan waktu. Besar nilai BER tidak dipengaruhi oleh waktu. Meskipun pada pengiriman sinyal informasi terkena *delay*, namun hal tersebut tidak memberikan pengaruh terhadap nilai BER yang dihasilkan. Hal tersebut dikarenakan BER bekerja pada *Layer 2* (*Datalink Layer*). Sedangkan untuk nilai *throughput*, besar nilai yang

dihasilkan dipengaruhi oleh waktu. Sehingga apabila pada saat pengiriman sinyal informasi terkena *delay*, maka hal tersebut akan memberikan pengaruh terhadap nilai *throughput* yang dihasilkan. Hal tersebut dikarenakan *throughput* bekerja pada *Layer 3 (Network Layer)*.

Pada praktiknya, tidak semua bit informasi yang diterima sama dengan bit informasi yang dikirim. Hal tersebut dipengaruhi oleh banyak faktor seperti suhu, *obstacle*, dan sebagainya. Perhitungan dari BER adalah [6]

$$BER = \frac{\text{Jumlah bit yang salah}}{\text{Jumlah bit yang dikirim}} \quad (2.18)$$

*Throughput* merupakan besar waktu yang ditempuh dari bit yang berhasil diterima penerima dengan benar. *Throughput* dapat dirumuskan sebagai berikut [12].

$$Throughput = \frac{\text{Jumlah bit yang benar}}{\text{Jumlah waktu transmisi}} \quad (2.19)$$





## BAB III

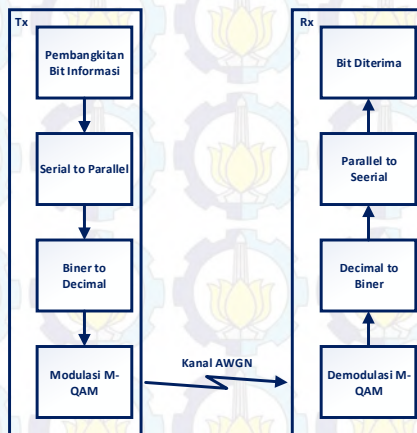
### PERANCANGAN DAN IMPLEMENTASI SISTEM

Bab III menjabarkan mengenai perancangan sistem, seperti langkah-langkah pembuatan *listing* program simulasi sebagai acuan awal sebelum diimplementasikan ke WARP. Selanjutnya, pada bab ini juga akan dijabarkan mengenai implementasi sistem. Implementasi sistem yang dimaksud, seperti integrasi PC dengan modul WARP, konfigurasi WARP, sampai dengan implementasi *listing* program yang telah disimulasikan ke modul WARP.

#### 3.1 Perancangan Simulasi Sistem pada Matlab

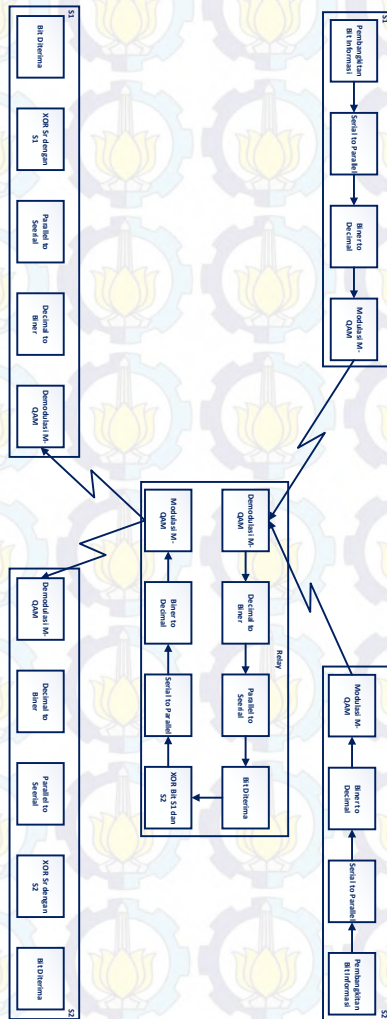
Sebelum melakukan implementasi sistem, sistem perlu diuji coba dengan simulasi. Simulasi ini dijadikan sebagai acuan awal dari implementasi sistem dan analisis data hasil dari implementasi sistem nantinya, selain bersumber dari landasan teori. Berikut merupakan blok diagram dari proses kerja simulasi sistem.

Gambar 3.1 merupakan proses sistem komunikasi M-QAM tradisional, yaitu sistem komunikasi *point to point*. *Network Coding* merupakan sistem komunikasi *relay*, sistem komunikasi dimana *node*



**Gambar 3.1** Blok Diagram Proses Simulasi Sistem Komunikasi M-QAM Traditional

asal dan tujuan dihubungkan oleh *relay*. Sehingga proses simulasi sistem komunikasi M-QAM dengan *Network Coding* digambarkan pada Gambar 3.2.



**Gambar 3.2** Blok Diagram Proses Simulasi Sistem Komunikasi M-QAM dengan *Network Coding*

Pada setiap *node* awal, S1 maupun S2, bit informasi dibangkitkan secara acak, deretan bit dapat dinyatakan sebagai

$$u(n) = [(1), (2), (3), \dots, (n)] \quad (3.1)$$

untuk  $n$  adalah banyak bit yang dibangkitkan secara acak. Deretan bit serial kemudian dibentuk menjadi deretan bit paralel sebanyak  $k$  paralel, dengan ukuran konstelasi M-QAM (Persamaan 2.1), dan dapat dinyatakan sebagai:

$$v = \frac{u(n)}{k} \quad (3.2)$$

Deretan bit paralel tersebut selanjutnya dikelompokkan menjadi simbol biner. Pada matlab 2013b, simbol yang dimodulasikan berupa simbol desimal. Oleh Karena itu, simbol biner diubah menjadi simbol desimal terlebih dahulu lalu dimodulasikan dengan modulasi QAM.

Pada program simulasi, kanal yang digunakan adalah kanal AWGN. Kanal AWGN ini untuk menggantikan kondisi kanal pada saat implementasi.

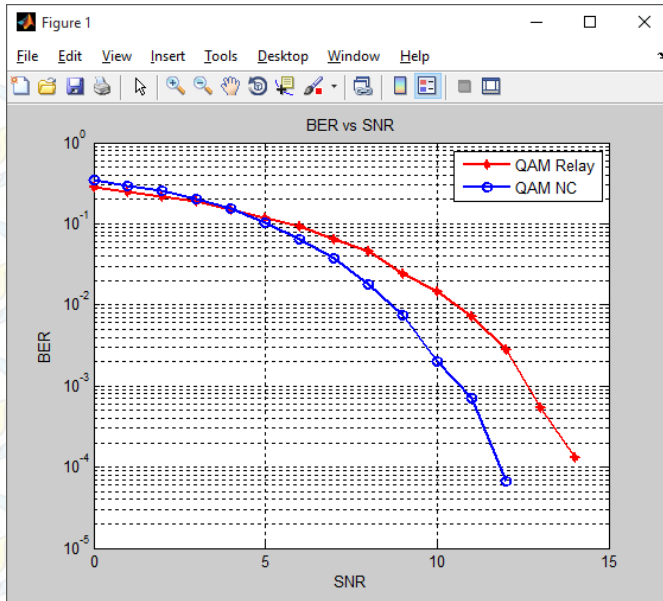
Sinyal dari S1 dan S2 di kirim ke *relay*. Sampai di penerima *relay*, sinyal yang telah melalui kanal AWGN didemodulasi dengan demodulasi QAM. Hasil demodulasi diubah kembali ke dalam bentuk biner lalu deretan biner paralel tersebut kembali dibentuk menjadi deretan bit serial. Deretan bit serial ini merupakan bit informasi yang dikirim dari *node* asal S1 maupun S2.

Bit-bit informasi yang telah diterima selanjutnya dilakukan proses *Network Coding*, seperti yang telah dijabarkan pada [7] bit informasi S1 dan S2 di-XOR. Hasil XOR kemudian dikirim kembali secara *broadcast* ke S1 dan S2 dengan proses pengiriman sama seperti pengirim S1 maupun S2.

Pada penerima S1 maupun S2, sinyal informasi dari *relay* didemodulasikan. Simbol desimal dari hasil demodulasi diubah kembali menjadi deretan paralel biner dan kemudian dibentuk menjadi deretan serial.

Di *node* S1, bit informasi yang diterima dari *relay* di-XOR-kan dengan bit informasi milik *node* S1 yang dikirimkan ke *node* S2 untuk mendapatkan bit informasi dari *node* S2. Proses yang sama juga dilakukan di *node* S2. Akan tetapi pada *node* S2 bit informasi dari *relay* di-OR-kan dengan bit informasi milik *node* S2. Gambar 3.3 merupakan gambar dari hasil simulasi.



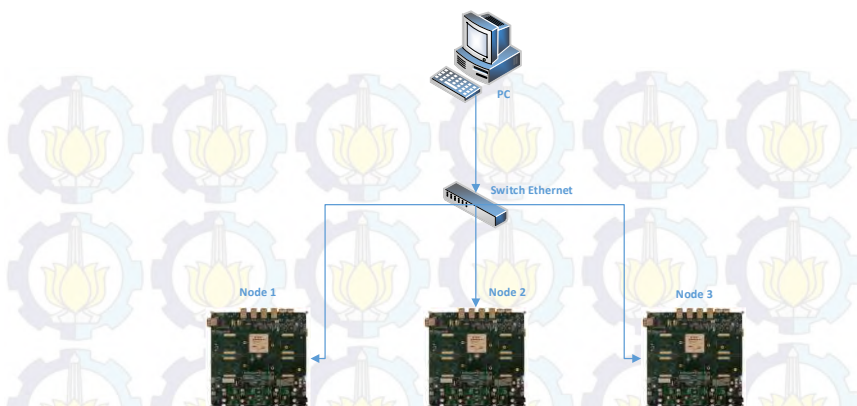


**Gambar 3.3** Hasil Simulasi BER 4 QAM *Relay* dan 4 QAM *Network Coding*

Dari Gambar 3.3, terdapat hasil simulasi dari QAM *Relay*. QAM *Relay* yang dimaksud yaitu sistem komunikasi relay dengan modulasi 4 QAM, namun tanpa ditambahkan dengan proses *network coding*. Sehingga Gambar 3.3 merupakan perbandingan BER antara sistem komunikasi *relay* dengan dan tanpa *network coding*. Hasil simulasi tersebut akan dijadikan sebagai acuan awal dalam implementasi sistem.

### 3.2 Integrasi PC dengan Modul WARP

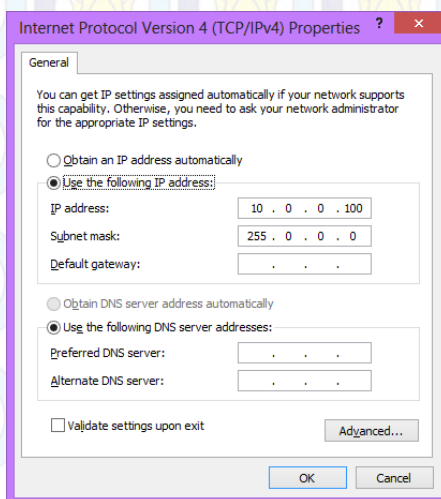
Pada penelitian Tugas Akhir ini, implementasi sistem menggunakan 3 modul WARP. Modul pertama (*node 1*) sebagai *node S1*, modul kedua (*node 2*) sebagai *node Relay*, dan modul 3 (*node 3*) sebagai *node S2*. Ketiga *node* tersebut diintegrasikan dengan satu unit PC dengan menggunakan satu buah *Switch* dan kabel LAN. Konfigurasi perangkat keras tersebut digambarkan pada Gambar 3.4.



**Gambar 3.4** Konfigurasi Integrasi PC dengan Modul WARP

### 3.2.1 Konfigurasi IP pada PC

PC dapat terintegrasi dengan modul WARP apabila alamat IP pada PC dan `warplab_defines` pada program matlab sama. Dalam pelaksanaannya, IP `warplab_defines` diatur mempunyai alamat IP 10.0.0.255. Sehingga alamat IP pada PC harus diatur menjadi 10.0.0.255 seperti digambarkan pada Gambar 3.5.



**Gambar 3.5** Pengaturan IP pada PC

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Windows\system32>arp -a

Interface: 10.0.0.100 --- Net
Internet Address      Physical Address      Type
10.122.125.1          00-1c-c5-06-c6-01    dynamic
10.255.255.255        ff-ff-ff-ff-ff-ff    static
224.0.0.2              01-00-5e-00-00-02    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-1b    static
224.0.0.252           01-00-5e-00-00-1c    static

C:\Windows\system32>arp -s 10.0.0.255 ff-ff-ff-ff-ff-ff

C:\Windows\system32>arp -a

Interface: 10.0.0.100 --- Net
Internet Address      Physical Address      Type
10.122.125.1          00-1c-c5-06-c6-01    dynamic
10.255.255.255        ff-ff-ff-ff-ff-ff    static
224.0.0.2              01-00-5e-00-00-02    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-1b    static
224.0.0.252           01-00-5e-00-00-1c    static

C:\Windows\system32>
```

**Gambar 3.6** Konfigurasi IP ARP pada CMD

Alamat IP PC yang sudah diatur, kemudian ditambahkan ke dalam tabel ARP. Alamat IP ditambahkan pada “*Command Prompt*” dengan langkah-langkah sebagai berikut:

- arp -a → untuk melihat alamat IP protocol dalam tabel ARP.
- arp -s 10.0.0.255 ff-ff-ff-ff-ff-ff → menambahkan IP pada tabel ARP.

langkah-langkah diatas digambarkan pada Gambar 3.6. Untuk memastikan bahwa alamat IP PC telah berhasil dimasukan ke dalam tabel ARP, dapat dilihat kembali ke dalam tabel ARP dengan mengetikan perintah “arp -a”.

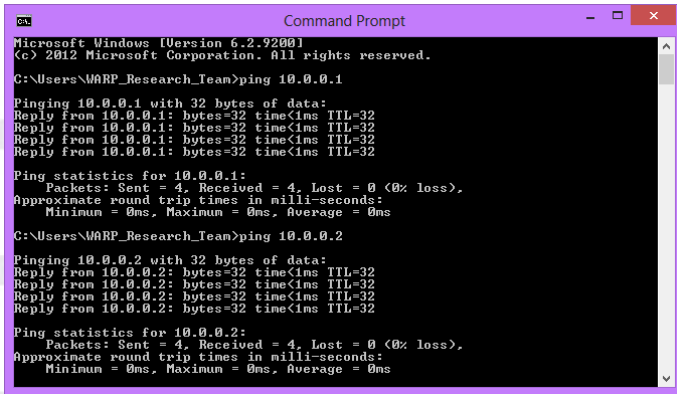
### 3.2.2 Tes Koneksi antara PC dan Modul WARP

IP pada PC yang telah diatur dan dimasukan ke dalam tabel ARP, kemudian dilakukan pengetesan koneksi antara PC dengan *node*. Tes koneksi dapat dilakukan melalui *Command Prompt* dan melalui Matlab.

#### 3.2.2.1 Cek Koneksi pada Command Prompt

Melalui *command prompt*, cek koneksi dapat menggunakan perintah “Ping”. Pada jendela *command prompt* masukan perintah ping lalu dilanjutkan dengan alamat IP dari masing-masing *node*. Perintah ping 10.0.0.1 untuk *node* 1, ping 10.0.0.2 untuk *node* 2, dan ping 10.0.0.3 untuk *node* 3. Indikasi bahwa





```
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. All rights reserved.

C:\Users\WARP_Research_Team>ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32
Reply from 10.0.0.1: bytes=32 time<1ms TTL=32

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\WARP_Research_Team>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:
Reply from 10.0.0.2: bytes=32 time<1ms TTL=32
Reply from 10.0.0.2: bytes=32 time<1ms TTL=32
Reply from 10.0.0.2: bytes=32 time<1ms TTL=32
Reply from 10.0.0.2: bytes=32 time<1ms TTL=32

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

**Gambar 3.7** Cek Koneksi Melalui *Command Prompt*

koneksi antara PC dan semua *node* telah terhubung diperlihatkan pada Gambar 3.7.

### 3.2.2.2 *Cek Koneksi pada Matlab*

Melalui matlab, cek koneksi dapat menggunakan baris perintah yang sudah ada dari *framework* WARPLab yang kita gunakan. Pada saat memasukan perintah ke *Command Window*, direktori target folder harus ditujukan ke folder “M\_Code\_Reference” dari *software* WARPLab versi 6/7. Untuk melakukan *setup* perangkat lunak pada modul WARP dapat memasukan perintah *wl\_setup*. Sedangkan untuk malakukan cek koneksi antara PC dan semua modul WARP dapat memasukan perintah *wl\_initNodes(x)*, untuk *x* adalah jumlah *node* yang digunakan. Gambar 3.8 dan Gambar 3.9 merupakan hasil dari *setup* dan cek koneksi pada matlab.

```

Command Window
>> wl_setup
Setting up WARP Lab Paths...
    removing path 'C:\WARP Lab7\M_Code_Reference\classes'
    removing path 'C:\WARP Lab7\M_Code_Reference\util'
    removing path 'C:\WARP Lab7\M_Code_Reference\config'
    adding path 'C:\WARP Lab7\M_Code_Reference\classes'
    adding path 'C:\WARP Lab7\M_Code_Reference\util'
    adding path 'C:\WARP Lab7\M_Code_Reference\util\inifile'
    adding path 'C:\WARP Lab7\M_Code_Reference\config'
    saving path

A wl_config.ini file was found in your path. Values specified in this
configuration file will be used as defaults in the construction of the
new file.

-----
Please enter a WARP Lab Ethernet interface address.

Pressing enter without typing an input will use a default
IP address of: 10.0.0.100

f WARP Lab Ethernet Interface Address: |

```

**Gambar 3.8** Hasil dari *setup* WARP pada Matlab

```

Command Window
>> wl_initNodes(3)

ans =

Displaying properties of 3 wl_node objects:
| ID | WLVER | HWVER | Serial # | Ethernet MAC Addr | Address |
|-----|-----|-----|-----|-----|-----|
| 0 | 7.1.0 | 2 | N/A | 00-50-C2-63-30-00 | 10.0.0.1 |
| 1 | 7.1.0 | 2 | N/A | 00-50-C2-63-30-01 | 10.0.0.2 |
| 2 | 7.1.0 | 2 | N/A | 00-50-C2-63-30-02 | 10.0.0.3 |

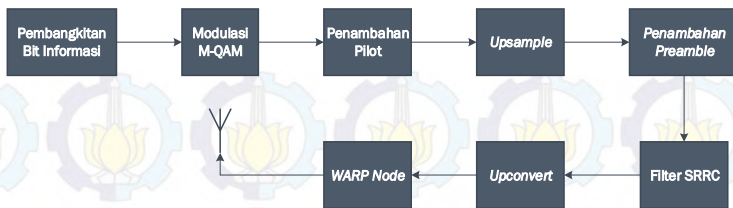
f >>

```

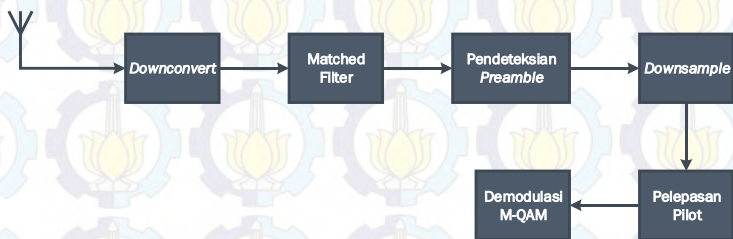
**Gambar 3.9** Hasil dari cek koneksi pada Matlab

### 3.3 Implementasi Sistem pada Modul WARP

Pada tahap implementasi, *listing* program yang dibuat untuk implementasi berbeda dengan *listing* program untuk simulasi. Letak perbedaannya yaitu pada pemodelan sistem implementasi yang meliputi pemodelan perangkat WARP beserta parameter-parameter dasar dalam implementasi WARP. Sehingga dilakukan perancangan kembali untuk sistem yang diimplementasikan ke modul WARP. Proses kerja dari sistem tersebut digambarkan pada blok diagram pada Gambar 3.10 dan Gambar 3.11.



**Gambar 3.10** Blok Diagram Proses Kerja pada Pengirim



**Gambar 3.11** Blok Diagram Proses Kerja pada Penerima

**Tabel 3.1** Parameter Sistem Implementasi WARP

No.	Parameter	Keterangan/Nilai
1	Level Modulasi QAM	4, 8, 16, 32, dan 64
2	Jumlah bit dikirim	30000
3	<i>Filter</i>	<i>Squared Root Raised Cosine</i>
4	Orde Filter	6
5	Faktor <i>Rolloff</i>	0.9
6	Jumlah <i>Preamble</i>	19
7	Jumlah <i>Corr Window</i>	300
8	Jumlah <i>Delay</i>	400
9	Frekuensi <i>Upconvert</i>	5 MHz
10	Frekuensi <i>Sampling</i>	40 MHz
11	<i>Carrier Channel</i>	Disesuaikan
12	<i>Upsample Rate</i>	13 <i>sample</i>
13	<i>Channel Estimator (CSI)</i>	150 simbol
14	<i>Symbol Rate</i>	$3 \times 10^6$
15	<i>Trasmitter Gain Baseband</i>	3
16	<i>Transmitter Gain RF</i>	0-63
17	<i>Receiver Gain Baseband</i>	1
18	<i>Receiver Gain RF</i>	1



### 3.3.1 Parameter Sistem

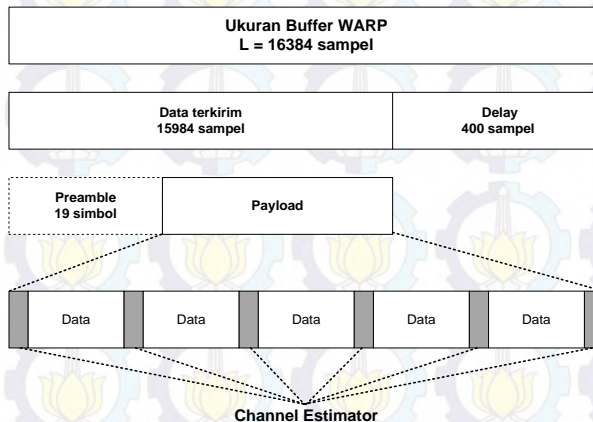
Pada perancangan sistem implementasi terdapat beberapa parameter dasar yang digunakan selama implementasi di WARP. Parameter tersebut diatur pada program matlab. Parameter-parameter tersebut dijabarkan pada Tabel 3.1 di bawah ini.

### 3.3.2 Desain Frame

Mengacu pada [13], setiap *buffer* baik pada Tx maupun Rx dapat menyimpan maksimal  $2^{14}$  sampel atau 16384 sampel dengan frekuensi *sampling* 40 MHz. Karena karakteristik dari implementasi ini melalui *Ethernet*, maka dalam keseluruhan *frame* harus dialokasikan beberapa sampel untuk *delay*. *Delay* akan diletakan di akhir *frame* dengan nilai 0, sehingga *delay* ini sering disebut juga sebagai *zero padding*. *Delay* pada *frame* ini dialokasikan sebesar 400 sampel. Secara keseluruhan, desain *frame* komunikasi dapat digambarkan pada Gambar 3.12.

### 3.3.3 Perancangan Sistem di Pemancar

Berdasarkan blok diagram pada Gambar 3.10 diatas, penjabaran dari proses kerja pada bagian pemancar adalah sebagai berikut.



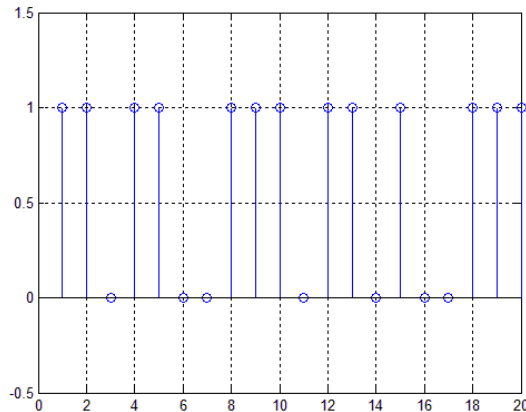
**Gambar 3.12** Struktur *Frame* Implementasi Sistem pada WARP

### 3.3.3.1 Pembangkitan Data Informasi

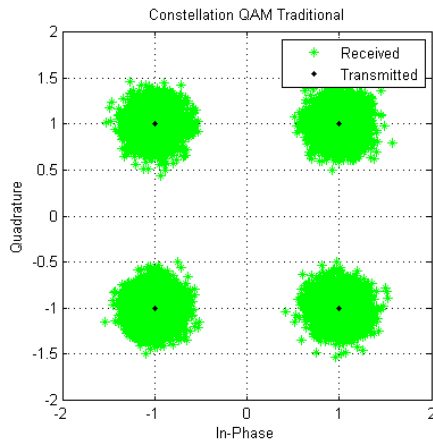
Bit data informasi yang dikirimkan sejumlah 30000 bit. Bit data informasi tersebut dibangkitkan secara acak dengan nilai 0 dan 1. Gambar 3.13 menunjukkan 20 bit pertama yang dibangkitkan dari pembangkitan bit acak dengan nilai 0 dan 1.

### 3.3.3.2 Modulasi M-QAM

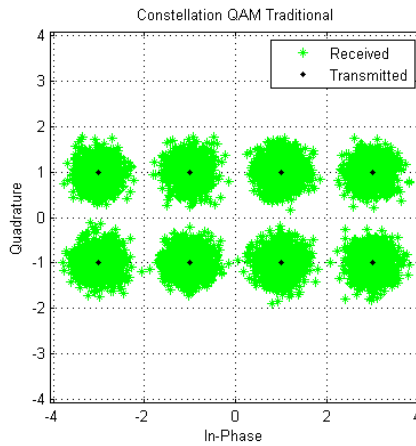
Bit informasi yang dihasilkan secara acak merupakan deretan bit dalam bentuk serial. Sebelum bit-bit informasi tersebut dimodulasi dengan modulasi QAM, bit-bit tersebut dikonversi menjadi deretan bit paralel sebanyak  $\log_2 M$ , dengan nilai M mempresentasikan jumlah bit data yang diwakili oleh satu simbol. Deretan bit paralel tersebut kemudian dimodulasi dengan modulasi QAM. Pada tugas akhir ini digunakan 5 modulasi yaitu, 4-QAM, 8-QAM, 16-QAM, 32-QAM, dan 64-QAM. Hasil modulasi dari kelima modulasi ditunjukkan pada Gambar 3.14, Gambar 3.15, Gambar 3.16, Gambar 3.17, dan Gambar 3.18.



**Gambar 3.13** Bit Informasi yang Dibangkitkan

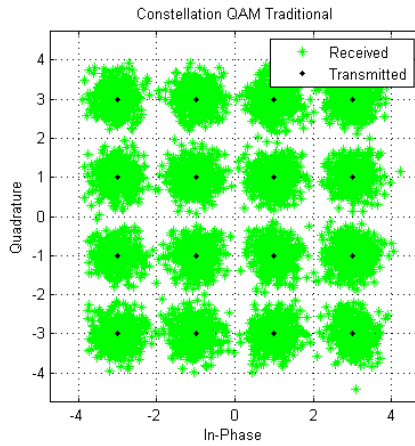


**Gambar 3.14** Konstelasi Simbol pada Level Modulasi 4-QAM

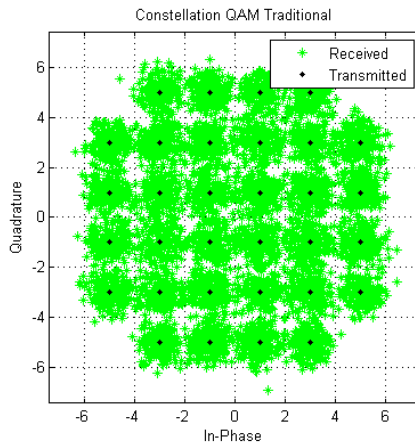


**Gambar 3.15** Konstelasi Simbol pada Level Modulasi 8-QAM

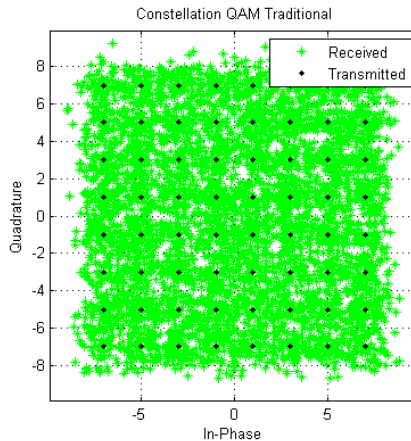




**Gambar 3.16** Konstelasi Simbol pada Level Moodulasi 16-QAM



**Gambar 3.17** Konstelasi Simbol pada Level Modulasi 32-QAM



**Gambar 3.18** Konstelasi Simbol pada Level Modulasi 64-QAM

### 3.3.3.3 Penambahan Pilot

Dalam proses pentransmisian data melalui kanal *wireless*, sinyal yang diterima biasanya mengalami kerusakan sinyal karena pengaruh kondisi kanal. Oleh karena itu, kondisi kanal harus diestimasi untuk mendapat kembali sinyal yang telah dikirim yaitu dengan menambahkan pilot disisi pengirim.

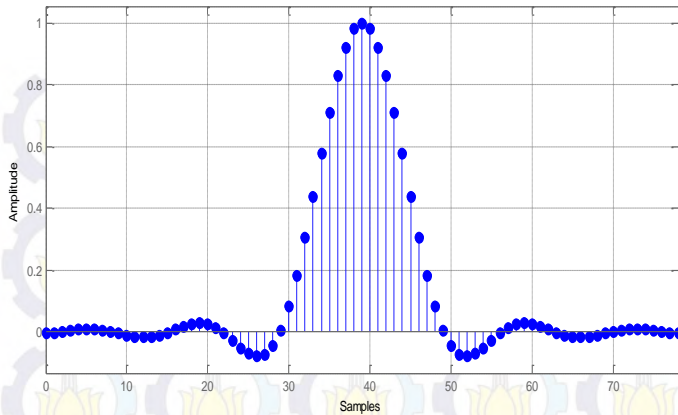
Berdasarkan permasalahan diatas, setelah proses modulasi kemudian dilakukan penambahan pilot. Simbol pilot disisipkan secara periodik. Pada tugas akhir ini digunakan simbol dengan nilai 1. Satu simbol pilot disisipkan untuk mengestimasi data simbol informasi dengan rasio 1/8. Sehingga setiap 8 bit simbol terdiri dari 1 bit yang berisi pilot dan 7 bit simbol yang berisi data. Desain penyusunan pilot dapat dilihat pada Gambar 3.19.

### 3.3.3.4 Upsample dan Penambahan Preamble

Proses selanjutnya adalah *upsample*. *Upsample* bertujuan untuk meningkatkan *sampling rate* dari data yang dikirim sesuai

$P_0$	$S_0 - S_6$	$P_1$	$S_7 - S_{13}$	...	$P_t$	$S_{n-7} - S_n$
-------	-------------	-------	----------------	-----	-------	-----------------

**Gambar 3.19** Susunan Pilot dan Simbol



**Gambar 3.20** Respon *Impuls* Filter *Squared Root Raised Cosine*

dengan jumlah *sample* yang digunakan sehingga diperoleh sinyal baru hasil *sampling*. Proses *upsample* ini dilakukan dengan menyisipkan bit bernilai 0 diantara bit-bit simbol informasi.

Penambahan *preamble* dimaksudkan untuk sinkronisasi transmisi dengan menggunakan deretan bit. Deretan bit yang digunakan tersebut telah diketahui oleh sistem dan kemudian diletakan pada awal deretan bit informasi yang dikirim.

### 3.3.3.5 *Filter SRRC*

Kemudian hasil *sampling* difilter dengan menggunakan filter *Squared Root Raised Cosine* (SRRC). Respon impuls dari filter SRRC ditunjukkan pada Gambar 3.20.

### 3.3.3.6 *Upconvert*

Proses *upconvert* digunakan untuk menggeser frekuensi *baseband* ke frekuensi radio. Pada [10], modul WARP bekerja pada frekuensi WiFi 2.4 GHz pada kanal 1-14. Frekuensi kerja sinyal dari hasil proses sebelumnya masih dalam frekuensi *baseband*, yaitu pada frekuensi 40 MHz. Oleh karena itu, sebelum sinyal dikirimkan, sinyal di-*upconvert* ke frekuensi radio 2.4 GHz.



### 3.3.4 Perancangan Sistem di WARP Transmit

Proses ini bertujuan untuk memberikan rute pengiriman kepada sinyal informasi yang dihasilkan oleh pengirim dan inisialisasi kepada semua *node*. *Node* manakah yang menjadi pengirim dan manakah yang menjadi penerima. Pada perancangan ini juga ditentukan pada kanal berapa pada frekuensi 2.4 GHz, frekuensi *node* bekerja dan ditentukannya besar *gain* pada pengirim dan penerima, pada frekuensi *baseband* dan frekuensi radio.

### 3.3.5 Perancangan Sistem di Penerima

Berdasarkan blok diagram pada Gambar 3.11 diatas, proses yang ada di penerima terbalik dari yang ada di pengirim. Penjabaran dari proses kerja pada bagian penerima adalah sebagai berikut.

#### 3.3.5.1 *Downconvert*

Pada blok penerima, awal proses dimulai dari proses *downconvert*. Proses *downconvert* mengembalikan frekuensi radio 2.4 GHz ke frekuensi *baseband* 40 MHz.

#### 3.3.5.2 *Matched Filter*

Berikutnya sinyal yang telah di-*downconvert* dilewatkan *matched filter*. Proses *matched filtering* menghilangkan pengaruh derau yang ada pada sinyal, dengan cara menghilangkan elemen frekuensi tinggi agar sesuai dengan bentuk pulsa *raised cosine* pada sisi pengirim.

#### 3.3.5.3 *Pendeteksian Preamble dan Downsample*

Pada proses ini akan dilakukan pendeteksian simbol pertama menggunakan *preamble*. Pendeteksian simbol pertama dilakukan berdasarkan korelasi tertinggi terhadap *preamble*.

Selanjutnya sinyal di-*downsample* dengan jumlah sampel sesuai dengan jumlah sampel pada proses *upsample*. Proses *downsample* menghilangkan penyisipan bit bernilai 0 diantara setiap kolom.

#### 3.3.5.4 *Pelepasan Pilot*

Hasil *downsample* kemudian dilakukan penghapusan simbol pilot yang sebelumnya disisipkan di pengirim.

### 3.3.5.5 Demodulasi M-QAM

Setelah proses diatas, selanjutnya sinyal didemodulasi dengan demodulasi QAM. Sinyal tersebut didemodulasi dengan konstelasi yang sesuai dengan konstelasi modulasi di pengirim. Hasil demodulasi berupa deretan bit paralel. Deretan paralel bit ini dikonversi menjadi deretan bit serial untuk mendapatkan bit informasi yang diinginkan.

### 3.3.6 Perancangan Sistem Network Coding

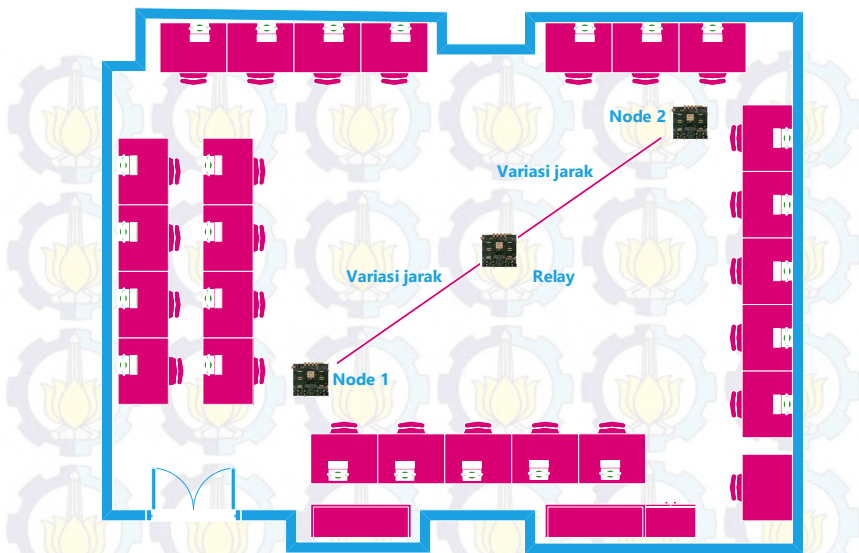
Sistem dari *network coding* bekerja di *relay*. Berdasarkan Gambar 2.12, sinyal informasi dari pengirim *node 1* dan *node 2* diterima oleh penerima *relay*. Kedua sinyal informasi tersebut melewati berbagai proses pada penerima seperti yang dijabarkan pada sub bab sebelumnya. Bit informasi yang didapat dari kedua sinyal dilakukan proses *network coding*, yaitu kedua sinyal di-XOR, sesuai yang tertera pada [7]. Bit hasil *network coding* selanjutnya dikirim kembali ke kedua *node* (*node 1* dan *node 2*) melalui pengirim *relay*. Proses pengirim pada *relay* sama seperti yang telah dijabarkan pada sub bab sebelumnya.

## 3.4 Skenario Pengukuran

Pengukuran sistem terimplementasi WARP dilakukan dalam 5 konstelasi modulasi, yaitu 4-QAM, 8-QAM, 16-QAM, 32-QAM, dan 64-QAM. Lingkungan tempat pengukuran dilakukan pada dua kondisi yang berbeda, yaitu pada kondisi dalam ruangan (*indoor*) dan pada kondisi luar ruangan (*outdoor*) dengan jarak yang berbeda-beda. Parameter yang diukur meliputi *Bit Error Rate* (BER), *Throughput*, dan Diagram Konstelasi. Ketiga parameter tersebut yang akan dibandingkan pada tiga sistem komunikasi yaitu M-QAM *Traditional* (*point to point*), M-QAM *Relay*, dan M-QAM *Network Coding*. Dari skenario pengukuran tersebut dapat dijadikan analisis perbandingan kinerja dari sistem komunikasi *network coding*.

### 3.4.1 Lingkungan Dalam Ruangan (Indoor)

Pengukuran dengan kondisi dalam ruangan dilakukan di ruangan AJ. 403, Laboraturium Praktikum Dasar Sistem Telekomunikasi, Jurusan Teknik Elektro ITS. Ketiga *node* diletakan pada posisi satu garis lurus. Pengaturan variasi jarak diberlakukan untuk jarak antara *node 1* dan *node 2*. *Node 1* dan



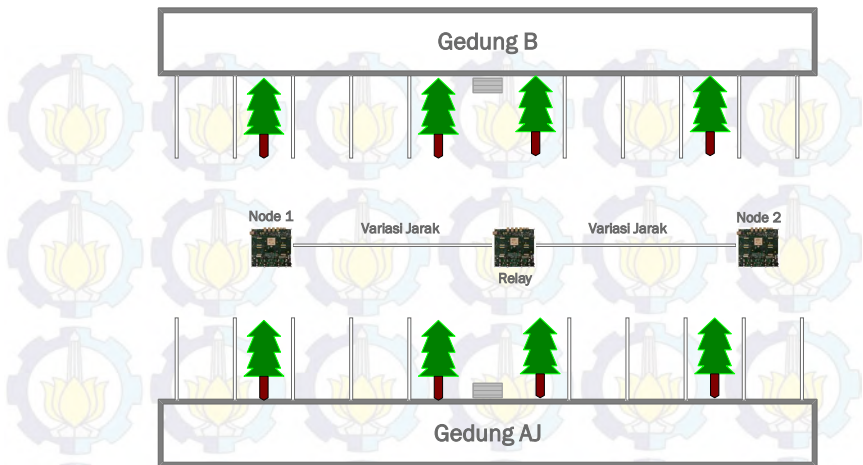
**Gambar 3.21** Konfigurasi Letak dan Jarak Tiap *Node* WARP Indoor

*node 2* merupakan *node* sumber sekaligus *node* tujuan, sedangkan *node relay* merupakan *node* yang menghubungkan jalur komunikasi antara *node 1* dan *node 2* pada komunikasi *relay* dan *network coding*. Konfigurasi letak dan jarak untuk tiap *node*-nya digambarkan pada Gambar 3.21.

### 3.4.2 Lingkungan Luar Ruangan (*Outdoor*)

Pengukuran dengan kondisi luar ruangan dilakukan di tempat parkir dosen Jurusan Teknik Elektro ITS. Ketiga *node* diletakkan pada posisi satu garis lurus. Pengaturan variasi jarak diberlakukan untuk jarak antara *node 1* dan *node 2*. *Node 1* dan *node 2* merupakan *node* sumber sekaligus *node* tujuan, sedangkan *node relay* merupakan *node* yang menghubungkan jalur komunikasi antara *node 1* dan *node 2* pada komunikasi *relay* dan *network coding*. Konfigurasi letak dan jarak untuk tiap *node*-nya digambarkan pada Gambar 3.22.





**Gambar 3.22** Konfigurasi Letak dan Jarak Tiap *Node WARP Outdoor*



## BAB IV

### HASIL DAN ANALISIS DATA

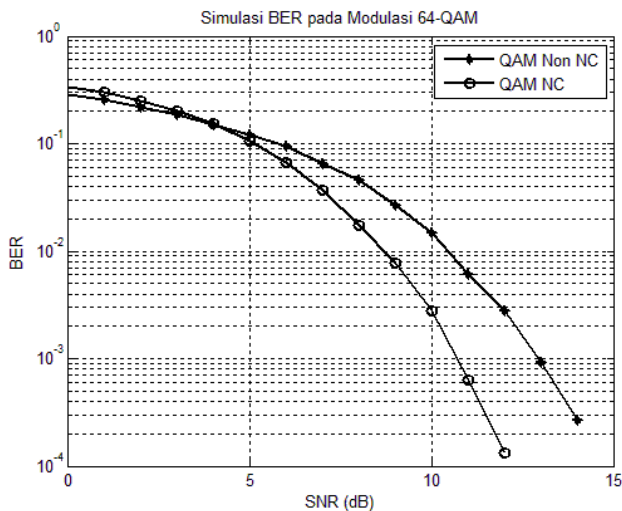
Pada bab ini semua hasil pengukuran dari implementasi sistem ke modul WARP dijabarkan dan kemudian dianalisis. Hasil dari simulasi juga dijabarkan sebagai pembandingan dengan hasil pengukuran. Mengacu pada [7], parameter yang diukur meliputi *Bit Error Rate* dan *Throughput*.

#### 4.1 Hasil Simulasi Sistem Modulasi M-QAM dengan Network Coding

Seperti yang telah dijabarkan pada bab III mengenai perancangan simulasi sistem. Hasil dari simulasi sistem bertujuan sebagai pembandingan atau sebagai acuan untuk hasil pengukuran agar tidak menghasilkan perbedaan yang signifikan. Hasil simulasi ditunjukkan pada gambar-gambar di bawah ini.

##### 4.1.1 Hasil Simulasi *Bit Error Rate* (BER)

Gambar 4.1 menggambarkan hasil simulasi kurva *Bit Error Rate* (BER) terhadap *Signal to Noise Ratio* (SNR) pada modulasi 4-QAM. Pada simulasi, untuk kondisi kanal digunakan kanal *Add*

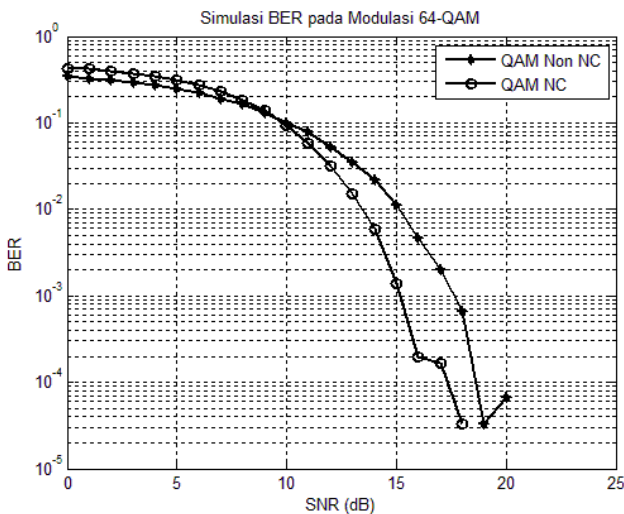


**Gambar 4.1** Hasil Simulasi BER pada Modulasi 4-QAM



*White Gaussian Noise to Signal* (AWGN). Penggunaan kanal AWGN pada simulasi bertujuan untuk mendapatkan kondisi kanal yang menyerupai kondisi kanal sebenarnya. Pada hasil diatas, nilai BER 4-QAM tanpa *network coding* dan BER 4-QAM dengan *network coding* sebelum SNR mencapai 4 dB, keduanya memiliki perbandingan yang tidak signifikan. Akan tetapi pada SNR 0 dB sampai dengan 3 dB, nilai BER 4-QAM dengan *network coding* lebih rendah dibandingkan dengan nilai BER 4-QAM tanpa *network coding*. Pada SNR 4 dB, nilai BER dari kedua sistem tersebut memiliki nilai yang hampir sama. Selanjutnya pada SNR yang lebih tinggi, nilai BER pada 4-QAM dengan *network coding* menjadi lebih baik dari nilai BER pada 4-QAM tanpa *network coding*. Dan semakin besar nilai SNR semakin besar pula perbandingan nilai BER-nya.

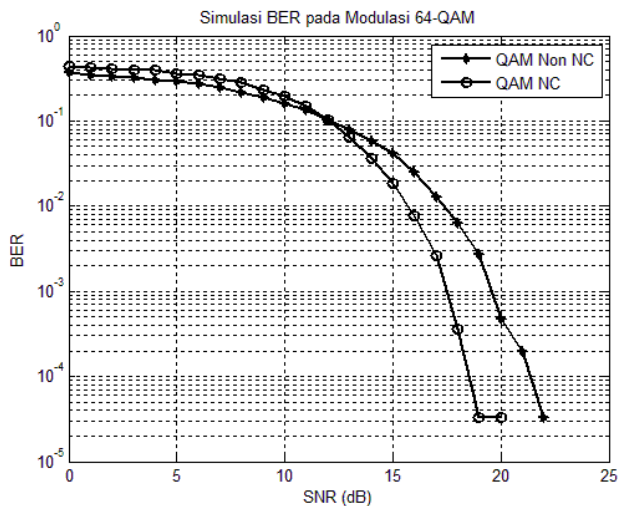
Gambar 4.2 menggambarkan hasil simulasi kurva *Bit Error Rate* (BER) terhadap *Signal to Noise Ratio* (SNR) pada modulasi 8-QAM. Sama halnya dengan hasil simulasi sebelumnya, pada simulasi ini juga digunakan kanal AWGN untuk mendapatkan kondisi kanal yang menyerupai kondisi kanal sebenarnya. Berbed



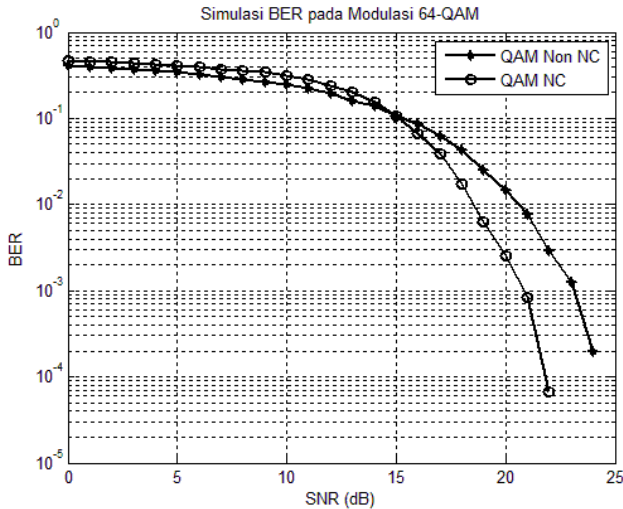
**Gambar 4.2** Hasil Simulasi BER pada Modulasi 8-QAM

dengan hasil simulasi 4-QAM, hasil simulasi 8-QAM menghasilkan kurva BER yang berbeda. Pada awal nilai SNR masih kurang dari 9 dB, nilai BER dari 8-QAM dengan *network coding* lebih rendah dari nilai BER dari 8-QAM tanpa *network coding*. Kurva dari kedua sistem terjadi perpotongan ketika SNR bernilai 9 dB. Dan seterusnya nilai BER dari 8-QAM dengan *network coding* semakin meningkat dan memberikan perbedaan nilai BER yang semakin besar mengikuti SNR yang semakin besar hingga bernilai 18 dB dan 19 dB.

Gambar 4.3 menggambarkan hasil simulasi kurva *Bit Error Rate* (BER) terhadap *Signal to Noise Ratio* (SNR) pada modulasi 16-QAM. Kanal yang digunakan yaitu kanal AWGN sama seperti hasil simulasi sebelum-sebelumnya. Pada kurva BER diatas tergambar bahwa posisi dimana nilai BER dari QAM dengan *network coding* lebih rendah dari QAM tanpa *network coding* lebih banyak dari pada modulasi 4-QAM dan 8-QAM. Nilai BER dari QAM dengan *network coding* tersebut menghasilkan nilai yang lebih rendah dari QAM tanpa *network coding* pada SNR 1 dB sampai dengan 12 dB. Pada SNR bernilai 13 dB terjadi



**Gambar 4.3** Hasil Simulasi BER pada Modulasi 16-QAM

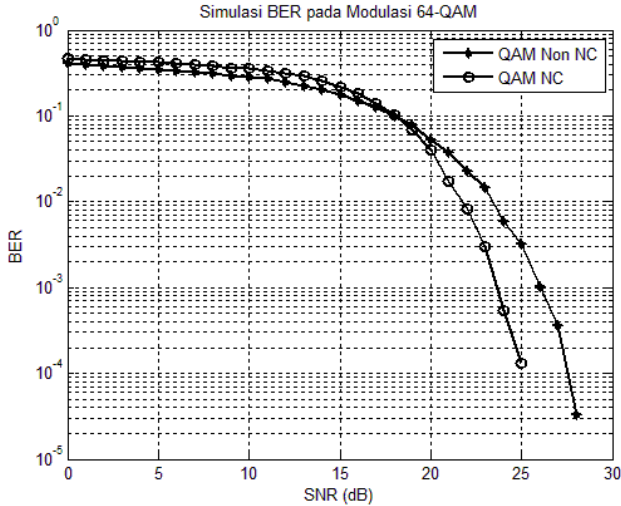


**Gambar 4.4** Hasil Simulasi BER pada Modulasi 32-QAM

perpotongan antara kedua kurva. Setelah perpotongan kurva tersebut nilai BER dari QAM dengan *network coding* meningkat menjadi lebih baik dari nilai BER QAM tanpa *network coding*. Selisih dari dua nilai BER tersebut semakin besar seiring dengan meningkatnya nilai SNR hingga mencapai 20 dB dan 23 dB.

Gambar 4.4 menggambarkan hasil simulasi kurva *Bit Error Rate* (BER) terhadap *Signal to Noise Ratio* (SNR) pada modulasi 32-QAM. Pada hasil modulasi 32-QAM, terdapat satu titik dimana dua kurva saling berpotongan, sama seperti kurva hasil simulasi sebelumnya. Yang membedakan dengan hasil sebelumnya, posisi dari titik dimana terjadi potongan antar kurva semakin bergeser ke nilai SNR yang lebih besar. Nilai BER dari QAM dengan *network coding* lebih rendah dari nilai BER dari QAM tanpa *network coding* pada saat SNR bernilai 1 hingga 15 dB. Panjang kondisi tersebut terjadi lebih panjang dari yang sebelumnya, yaitu pada SNR 1 dB hingga 12 dB. Setelah nilai SNR melewati 15 dB, nilai BER pada QAM dengan *network coding* menjadi lebih baik dibandingkan dengan nilai BER pada QAM tanpa *network coding*.





**Gambar 4.5** Hasil Simulasi BER pada Modulasi 64-QAM

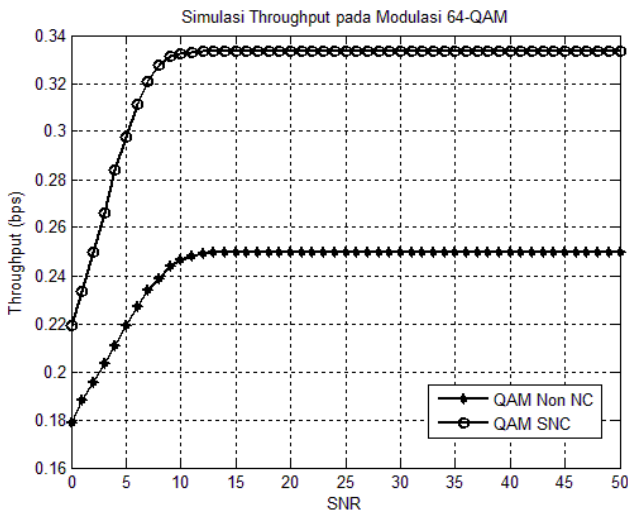
Gambar 4.5 menggambarkan hasil simulasi kurva *Bit Error Rate* (BER) terhadap *Signal to Noise Ratio* (SNR) pada modulasi 64-QAM. Diamati dari hasil simulasi diatas, pola yang sama berlaku kembali pada hasil simulasi ini. Pola yang dimaksud, yaitu di awal nilai BER dengan *network coding* lebih rendah daripada nilai BER tanpa *network coding*. Selanjutnya dua kurva saling berpotongan dan kemudian nilai BER dengan *network coding* menjadi lebih baik daripada nilai BER tanpa *network coding*. Jika diperhatikan kembali level modulasi dari QAM memberikan dampak pada titik perpotongan dua kurva di setiap hasil simulasi. Semakin besar nilai level modulasi maka titik perpotongan akan bergeser lebih besar pula. Pada hasil simulasi diatas, titik perpotongan bergeser dari yang sebelumnya pada modulasi 32-QAM terletak pada SNR 15 dB kini berada di 18 dB. Selain posisi titik perpotongan, semua terjadi seperti pola sebelumnya. Pada nilai SNR dibawah SNR titik potong, nilai BER pada QAM dengan *network coding* selalu lebih rendah dibanding dengan nilai BER pada QAM tanpa *network coding*. Setelah nilai SNR lebih besar dari SNR titik potong, maka nilai BER pada QAM dengan *network*

*coding* menjadi lebih baik dan terus meningkat hingga memberikan perbedaan yang signifikan diakhirnya.

#### 4.1.2 Hasil Simulasi *Throughput*

Gambar 4.6 merupakan penggambaran dari kurva *throughput* dari sistem QAM dengan *network coding* dan sistem QAM tanpa *network coding*. Pada simulasi ini, untuk mendekati dengan keadaan yang sebenarnya, waktu transmisi paket per kanal diasumsikan sebesar 1 detik. Besar nilai *throughput* diawal kurva untuk QAM tanpa *network coding* yaitu sebesar  $10^{-0.75}$  bps. Nilai *throughput* terus meningkat seiring meningkatnya nilai SNR sampai pada SNR 15 dB. Setelah dari 15 dB, nilai *throughput* tetap konstan pada  $10^{-0.6}$  bps.

Lalu untuk QAM dengan *network coding* memiliki bentuk kurva yang berbeda dengan QAM tanpa *network coding*. Dimulai dengan nilai *throughput* sebesar  $10^{-0.65}$  bps. Nilai *throughput* tersebut terus meningkat sampai SNR 12 dB. Setelah melebihi nilai SNR 12 dB maka nilai *throughput* dari QAM dengan *network coding* menjadi konstan pada nilai  $10^{-0.48}$  bps. Secara teori, sistem dengan *network coding* mampu memperbaiki *throughput* hingga



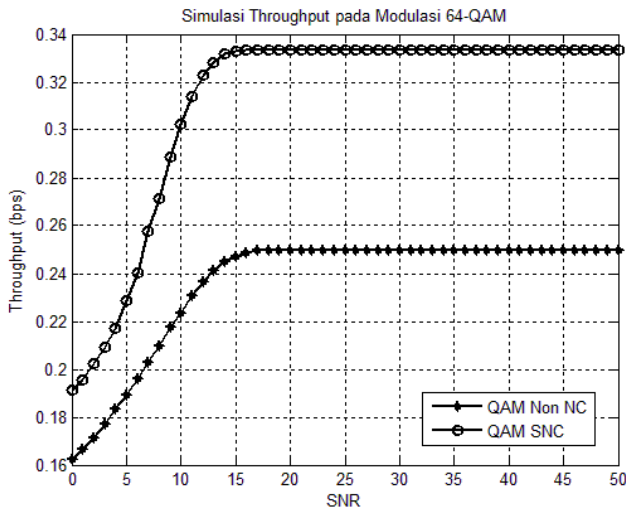
**Gambar 4.6** Hasil Simulasi *Throughput* pada Modulasi 4-QAM

33% [7]. Jika diamati lebih lanjut dapat dilakukan perhitungan sebagai berikut:

$$\frac{QAM\ NC - QAM\ Non\ NC}{QAM\ Non\ NC} \times 100\% = 31.8\%$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 31.8 %. Hasil tersebut hampir mendekati teori.

Gambar 4.7 merupakan penggambaran dari kurva *throughput* pada modulasi 8-QAM. Pengaturan yang diberikan pada hasil simulasi diatas sama seperti pengaturan pada simulasi dengan modulasi 4-QAM. Besar nilai *throughput* diawal kurva untuk QAM tanpa *network coding* yaitu sebesar  $10^{-0.78}$  bps. Nilai *throughput* terus meningkat seiring meningkatnya nilai SNR sampai pada SNR 19 dB. Setelah dari 19 dB, nilai *throughput* tetap konstan pada  $10^{-0.6}$  bps.



**Gambar 4.7** Hasil Simulasi *Throughput* pada Modulasi 8-QAM

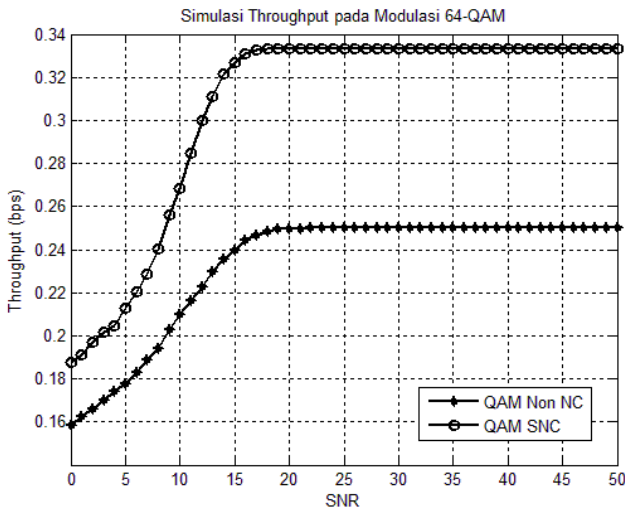


Lalu untuk QAM dengan *network coding* memiliki bentuk kurva yang berbeda dengan QAM tanpa *network coding*. Dimulai dengan nilai *throughput* sebesar  $10^{-0.71}$  bps. Nilai *throughput* tersebut terus meningkat sampai SNR 16 dB. Setelah melebihi nilai SNR 16 dB maka nilai *throughput* dari QAM dengan *network coding* menjadi konstan pada nilai  $10^{-0.48}$  bps. Secara teori, sistem dengan *network coding* mampu memperbaiki *throughput* hingga 33% [7]. Jika diamati lebih lanjut dapat dilakukan perhitungan sebagai berikut:

$$\frac{QAM\ NC - QAM\ Non\ NC}{QAM\ Non\ NC} \times 100\% = \frac{10^{-0.48} - 10^{-0.6}}{10^{-0.6}} \times 100\% = 31.8\%$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 31.8 %. Hasil tersebut hampir mendekati teori.

Gambar 4.8 merupakan penggambaran dari kurva *throughput* pada modulasi 16-QAM. Pengaturan yang diberikan pada hasil



**Gambar 4.8** Hasil Simulasi *Throughput* pada Modulasi 16-QAM

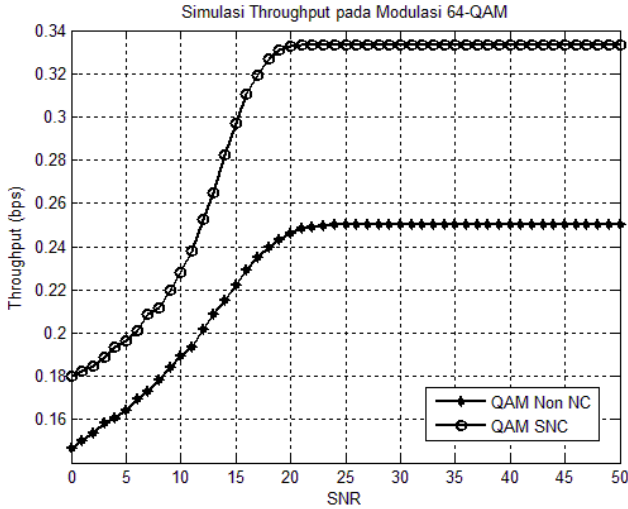
simulasi diatas sama seperti pengaturan pada simulasi dengan modulasi 4-QAM. Besar nilai *throughput* diawal kurva untuk QAM tanpa *network coding* yaitu sebesar  $10^{-0.8}$  bps. Nilai *throughput* terus meningkat seiring meningkatnya nilai SNR sampai pada SNR 20 dB. Setelah dari 20 dB, nilai *throughput* tetap konstan pada  $10^{-0.6}$  bps.

Lalu untuk QAM dengan *network coding* memiliki bentuk kurva yang berbeda dengan QAM tanpa *network coding*. Dimulai dengan nilai *throughput* sebesar  $10^{-0.72}$  bps. Nilai *throughput* tersebut terus meningkat sampai SNR 18 dB. Setelah melebihi nilai SNR 18 dB maka nilai *throughput* dari QAM dengan *network coding* menjadi konstan pada nilai  $10^{-0.48}$  bps. Secara teori, sistem dengan *network coding* mampu memperbaiki *throughput* hingga 33% [7]. Jika diamati lebih lanjut dapat dilakukan perhitungan sebagai berikut:

$$\frac{QAM\ NC - QAM\ Non\ NC}{QAM\ Non\ NC} = \frac{10^{-0.48} - 10^{-0.6}}{10^{-0.6}} = 0.08$$

$$\times 100\% = 31.8\ \%$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 31.8 %. Hasil tersebut hampir mendekati teori.



**Gambar 4.9** Hasil Simulasi *Throughput* pada Modulasi 32-QAM

Gambar 4.9 merupakan penggambaran dari kurva *throughput* pada modulasi 32-QAM. Pengaturan yang diberikan pada hasil simulasi diatas sama seperti pengaturan pada simulasi dengan modulasi 4-QAM. Besar nilai *throughput* diawal kurva untuk QAM tanpa *network coding* yaitu sebesar  $10^{-0.83}$  bps. Nilai *throughput* terus meningkat seiring meningkatnya nilai SNR sampai pada SNR 23 dB. Setelah dari 23 dB, nilai *throughput* tetap konstan pada  $10^{-0.6}$  bps.

Lalu untuk QAM dengan *network coding* memiliki bentuk kurva yang berbeda dengan QAM tanpa *network coding*. Dimulai dengan nilai *throughput* sebesar  $10^{-0.75}$  bps. Nilai *throughput* tersebut terus meningkat sampai SNR 21 dB. Setelah melebihi nilai SNR 21 dB maka nilai *throughput* dari QAM dengan *network coding* menjadi konstan pada nilai  $10^{-0.48}$  bps. Secara teori, sistem dengan *network coding* mampu memperbaiki *throughput* hingga 33% [7]. Jika diamati lebih lanjut dapat dilakukan perhitungan sebagai berikut:

$$QAM\ NC - QAM\ Non\ NC = 10^{-0.48} - 10^{-0.6} = 0.08$$

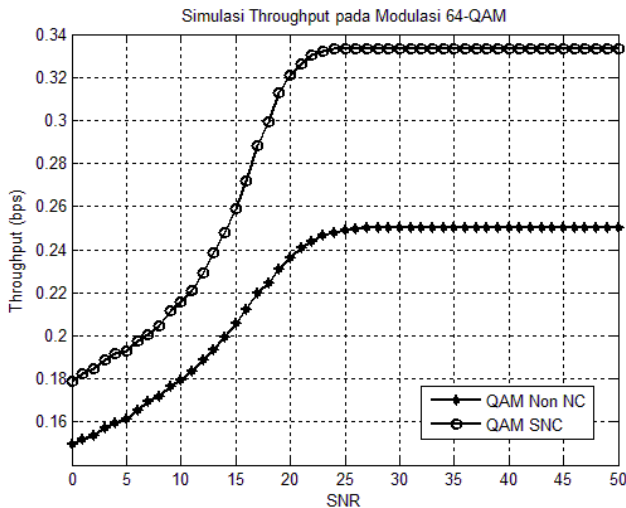


$$\frac{0.08}{QAM \text{ Non NC}} \times 100\% = 31.8 \%$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 31.8 %. Hasil tersebut hampir mendekati teori.

Gambar 4.10 merupakan penggambaran dari kurva *throughput* pada modulasi 64-QAM. Pengaturan yang diberikan pada hasil simulasi diatas sama seperti pengaturan pada simulasi dengan modulasi 4-QAM. Besar nilai *throughput* diawal kurva untuk QAM tanpa *network coding* yaitu sebesar  $10^{-0.83}$  bps. Nilai *throughput* terus meningkat seiring meningkatnya nilai SNR sampai pada SNR 27 dB. Setelah dari 27 dB, nilai *throughput* tetap konstan pada  $10^{-0.6}$  bps.

Lalu untuk QAM dengan *network coding* memiliki bentuk kurva yang berbeda dengan QAM tanpa *network coding*. Dimulai dengan nilai *throughput* sebesar  $10^{-0.75}$  bps. Nilai *throughput* tersebut terus meningkat sampai SNR 24 dB. Setelah melebihi nilai SNR 24 dB maka nilai *throughput* dari QAM dengan *network*



**Gambar 4.10** Hasil Simulasi *Throughput* pada Modulasi 64-QAM

*coding* menjadi konstan pada nilai  $10^{-0.48}$  bps. Secara teori, sistem dengan *network coding* mampu memperbaiki *throughput* hingga 33% [7]. Jika diamati lebih lanjut dapat dilakukan perhitungan sebagai berikut:

$$\begin{aligned} QAM\ NC - QAM\ Non\ NC &= 10^{-0.48} - 10^{-0.6} = 0.08 \\ \frac{0.08}{QAM\ Non\ NC} \times 100\% &= 31.8\ \% \end{aligned}$$

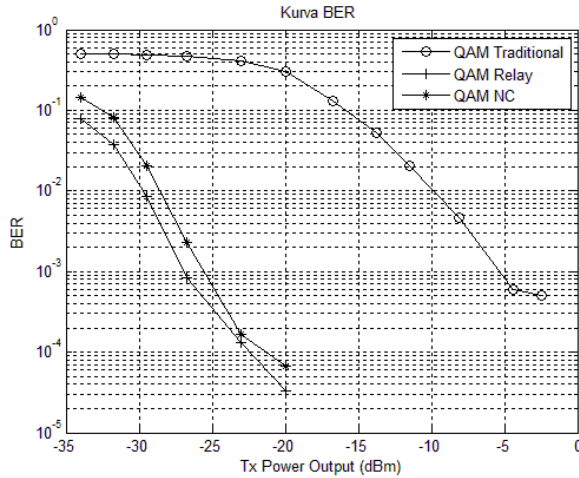
hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 31.8 %. Hasil tersebut hampir mendekati teori.

Dilihat dari kelima level modulasi, mereka mempunyai bentuk kurva yang hampir sama semua. Akan tetapi, tiap level memiliki nilai *throughput* di awal kurva yang berbeda-beda. Lalu di tiap level modulasi memiliki titik konstan yang berbeda-beda pula. Dan pada akhirnya persentasi perbaikan *throughput* yang diberikan simulasi sistem *network coding* mendekati teori.

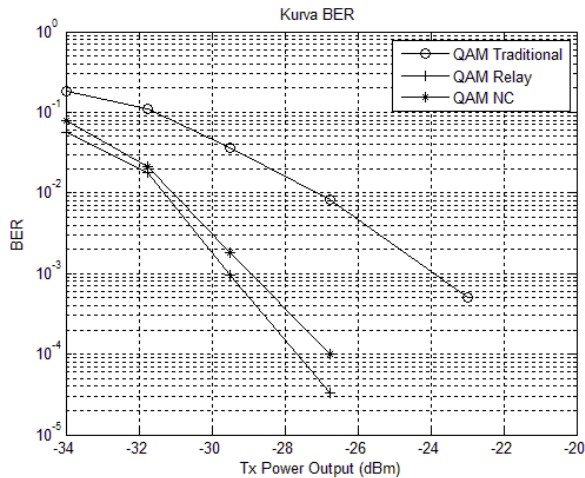
#### **4.2 Analisis Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point***

Pada pengukuran ini dilakukan pengamatan terhadap Sistem *Network Coding*, Sistem Tanpa *Network Coding*, dan Sistem *Point to Point* QAM. Pengamatan dilakukan dengan membandingkan kinerja dari ketiga sistem tersebut. Kinerja dari sebuah sistem dapat diamati dengan membandingkan dari beberapa parameter, seperti BER ataupun *Throughput*. Pengukuran dilakukan pada kondisi lingkungan dalam ruangan dan lingkungan luar ruangan. Hasil dari pengukuran yang digunakan sebagai analisis merupakan hasil pengukuran pada jarak 8 meter.

#### 4.2.1 Analisis Sistem pada Modulasi 4-QAM



**Gambar 4.11** Kurva BER 4-QAM *Indoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

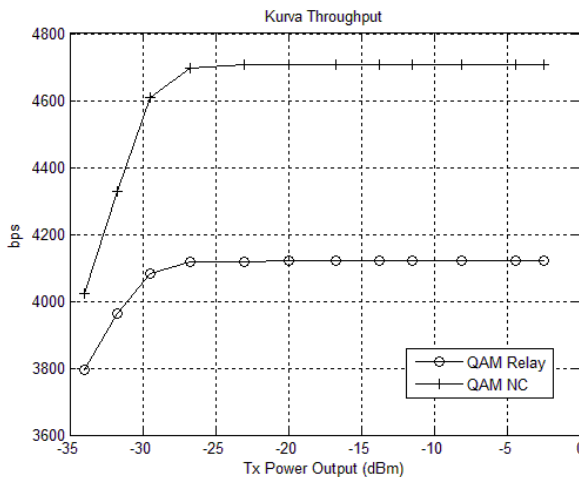


**Gambar 4.12** Kurva BER 4-QAM *Outdoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

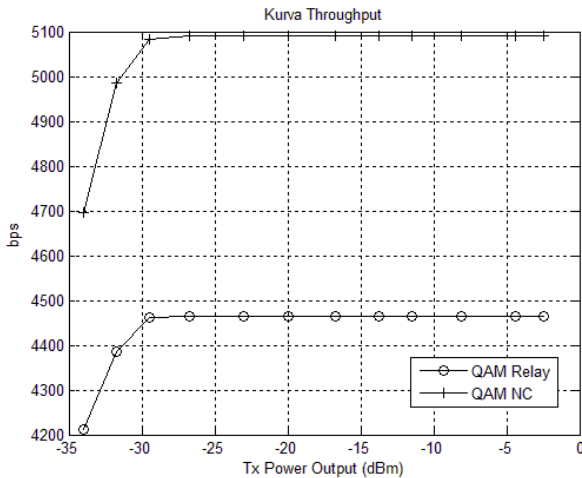


Hasil dari pengukuran perbandingan tiga sistem komunikasi, Sistem 4-QAM dengan *Network Coding*, 4-QAM tanpa *Network Coding*, dan 4-QAM *Point to Point* digambarkan pada gambar diatas. Metode pengukuran telah dijelaskan pada perancangan Bab III. Garis kurva pada gambar diatas yang memiliki nilai BER paling rendah baik pada kondisi *indoor* maupun *outdoor* yaitu BER dari 4-QAM *point to point*. Hasil tersebut dikarenakan pada sistem *point to point* tidak menggunakan *relay*. Pada sistem *point to point* sinyal dipancarkan langsung dari sumber 1 ke sumber 2. Sehingga daya sinyal yang diterima lebih lemah dibandingkan dengan daya sinyal pada sistem yang menggunakan *relay*.

Pada gambar diatas pada kondisi *indoor* dan *outdoor*, 4-QAM dengan *network coding* memiliki nilai BER yang lebih rendah dari nilai BER 4-QAM tanpa *network coding*. Meskipun 4-QAM dengan *network coding* memiliki nilai BER yang lebih rendah namun perbandingannya dengan nilai BER dari 4-QAM tanpa *network coding* tidak signifikan dan nilai BER keduanya pun tetap tidak memberikan selisih BER yang besar sampai akhirnya BER bernilai 0. Dari hasil tersebut telah menggambarkan bahwa hasil kurva diatas mendekati kurva pada simulasi dan pada [7].



**Gambar 4.13** Kurva *Throughput* 4-QAM *Indoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*



**Gambar 4.14** Kurva *Throughput* 4-QAM *Outdoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

Gambar diatas merupakan hasil pengukuran *throughput* dari sistem 4-QAM dengan *network coding* dan sistem 4-QAM tanpa *network coding*. Hasil pengukuran pada kondisi *indoor* dan pada kondisi *outdoor* memiliki bentuk kurva yang hampir mirip, yang membedakan dari kedua kurva tersebut terletak pada besar nilai *throughput*-nya. Pada kondisi *outdoor* nilai *throughput* yang dihasilkan lebih baik atau lebih tinggi dibanding dengan nilai yang dihasilkan pada kondisi *indoor*.

Pada kondisi lingkungan yang sama, baik pada kondisi *indoor* maupun pada kondisi *outdoor*, kurva dari 4-QAM dengan *network coding* menghasilkan nilai *throughput* yang jauh lebih baik dibandingkan dengan kurva dari 4-QAM tanpa *network coding*. Selisih dari keduanya menunjukan nilai selisih yang cukup signifikan. Ditinjau dari nilai maksimum *throughput* dari kedua sistem pada kondisi *indoor*, pada *power output* -16.75 dBm nilai *throughput* dengan *network coding* mencapai 4,7066 kbps sedangkan tanpa *network coding* 4,1197 kbps. Bila dilakukan perhitungan persentase dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

$$QAM\ NC - QAM\ Non\ NC = 4706.6 - 4119.7 = 586.9$$

$$\frac{586.9}{QAM\ Non\ NC} \times 100\% = 14.24\%$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 14,24 %.

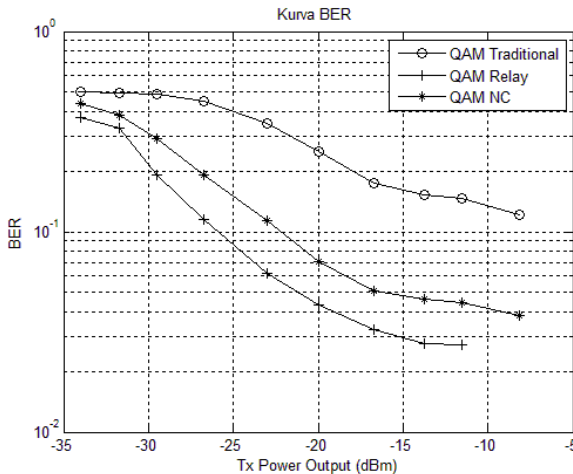
Dan bila ditinjau dari nilai maksimum *throughput* dari kedua sistem pada kondisi *outdoor*, pada *power output* -23 dBm nilai *throughput* dengan *network coding* mencapai 5,0918 kbps sedangkan tanpa *network coding* 4,4656 kbps. Bila dilakukan perhitungan persentase dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

$$QAM\ NC - QAM\ Non\ NC = 5091.8 - 4465.6 = 626.2$$

$$\frac{626.2}{QAM\ Non\ NC} \times 100\% = 14.02\%$$

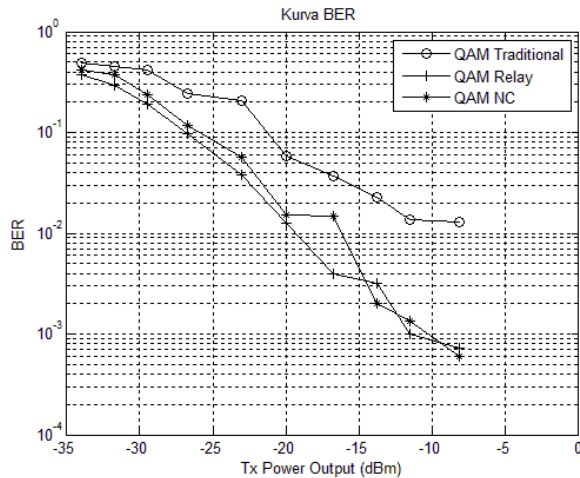
hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 14,02 %.

#### 4.2.2 Analisis Sistem pada Modulasi 8-QAM



**Gambar 4.15** Kurva BER 8-QAM Indoor dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

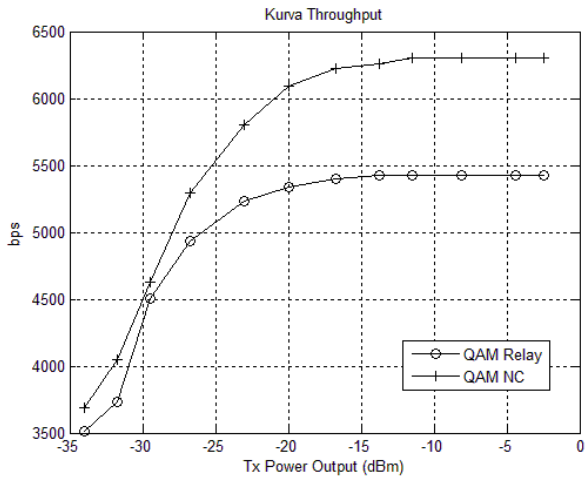




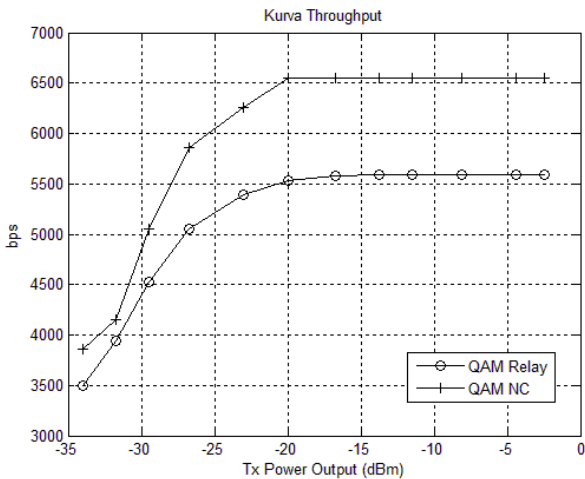
**Gambar 4.16** Kurva BER 8-QAM Outdoor dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

Hasil dari pengukuran perbandingan tiga sistem komunikasi, Sistem 8-QAM dengan *Network Coding*, 8-QAM tanpa *Network Coding*, dan 8-QAM *Point to Point* digambarkan pada gambar diatas. Garis kurva pada gambar diatas yang memiliki nilai BER paling rendah baik pada kondisi *indoor* maupun *outdoor* yaitu BER dari 8-QAM *point to point*. Hasil tersebut dikarenakan pada sistem *point to point* tidak menggunakan *relay*. Pada sistem *point to point* sinyal dipancarkan langsung dari sumber 1 ke sumber 2. Sehingga daya sinyal yang diterima lebih lemah dibandingkan dengan daya sinyal pada sistem yang menggunakan *relay*.

Pada gambar diatas pada kondisi *indoor* dan *outdoor*, 8-QAM dengan *network coding* memiliki nilai BER yang lebih rendah dari nilai BER 8-QAM tanpa *network coding*. Meskipun 8-QAM dengan *network coding* memiliki nilai BER yang lebih rendah namun perbedaannya dengan nilai BER dari 8-QAM tanpa *network coding* tidak signifikan dan nilai BER keduanya pun tetap tidak memberikan selisih BER yang besar sampai akhirnya BER bernilai 0. Dari hasil tersebut telah menggambarkan bahwa hasil kurva diatas mendekati kurva pada simulasi dan pada [7].



**Gambar 4.17** Kurva *Throughput* 8-QAM Indoor dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*



**Gambar 4.18** Kurva *Throughput* 8-QAM Outdoor dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

Gambar diatas merupakan hasil pengukuran *throughput* dari sistem 8-QAM dengan *network coding* dan sistem 8-QAM tanpa *network coding*. Hasil pengukuran pada kondisi *indoor* dan pada kondisi *outdoor* memiliki bentuk kurva yang hampir mirip, yang membedakan dari kedua kurva tersebut terletak pada besar nilai *throughput*-nya sama pada hasil modulasi 8-QAM. Pada kondisi *outdoor* nilai *throughput* yang dihasilkan lebih baik atau lebih tinggi dibanding dengan nilai yang dihasilkan pada kondisi *indoor*.

Pada kondisi lingkungan yang sama, baik pada kondisi *indoor* maupun pada kondisi *outdoor*, kurva dari 8-QAM dengan *network coding* menghasilkan nilai *throughput* yang jauh lebih baik dibandingkan dengan kurva dari 8-QAM tanpa *network coding*. Berbeda dengan hasil *throughput* pada modulasi 4-QAM, pada modulasi ini selisih dari keduanya menunjukan nilai selisih yang tidak besar. Ditinjau dari nilai maksimum *throughput* dari kedua sistem pada kondisi *indoor*, pada *power output* -11,5 dBm nilai *throughput* dengan *network coding* mencapai 6,3022 kbps sedangkan tanpa *network coding* 5,4282 kbps. Bila dilakukan perhitungan persentase dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

$$\begin{aligned} QAM\ NC - QAM\ Non\ NC &= 6302.2 - 5428.2 = 874 \\ \frac{874}{QAM\ Non\ NC} \times 100\% &= 16.1\% \end{aligned}$$

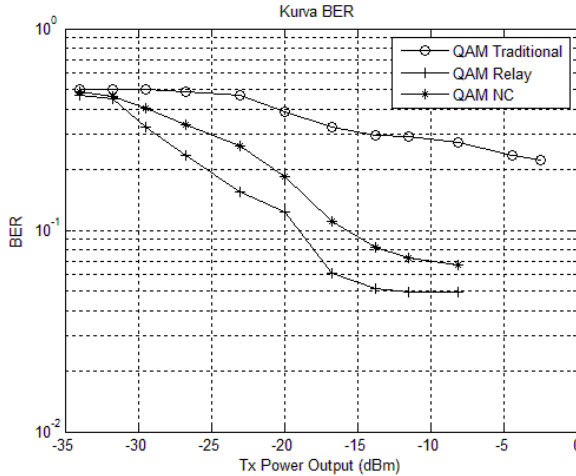
hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 16,1 %.

Dan bila ditinjau dari nilai maksimum *throughput* dari kedua sistem pada kondisi *outdoor*, pada *power output* -11,5 dBm nilai *throughput* dengan *network coding* mencapai 6,638 kbps sedangkan tanpa *network coding* 5,5936 kbps. Bila dilakukan perhitungan persentase dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

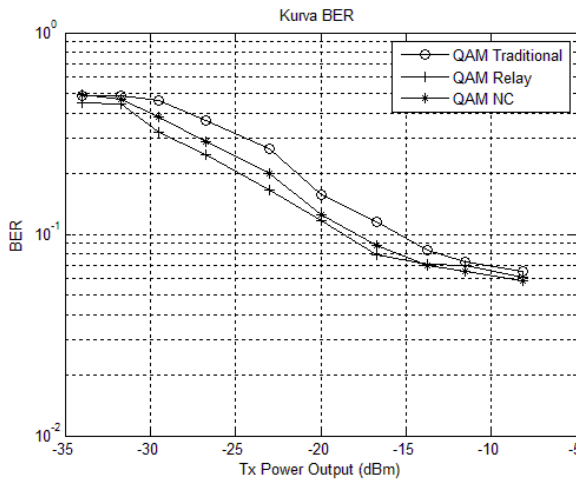
$$\begin{aligned} QAM\ NC - QAM\ Non\ NC &= 6638 - 5593.6 = 1044.4 \\ \frac{1044.4}{QAM\ Non\ NC} \times 100\% &= 18.67\% \end{aligned}$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 18.67 %.

### 4.2.3 Analisis Sistem pada Modulasi 16-QAM



**Gambar 4.19** Kurva BER 16-QAM *Indoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

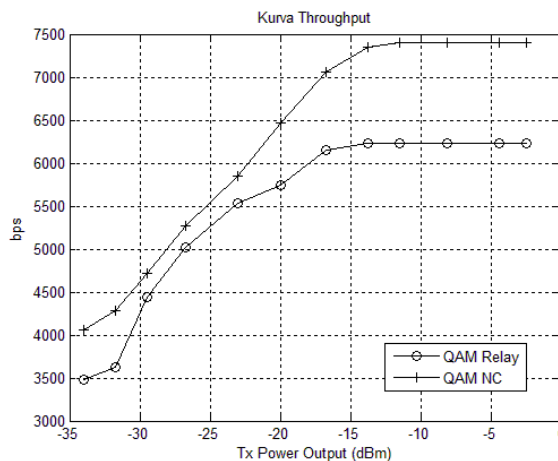


**Gambar 4.20** Kurva BER 16-QAM *Outdoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

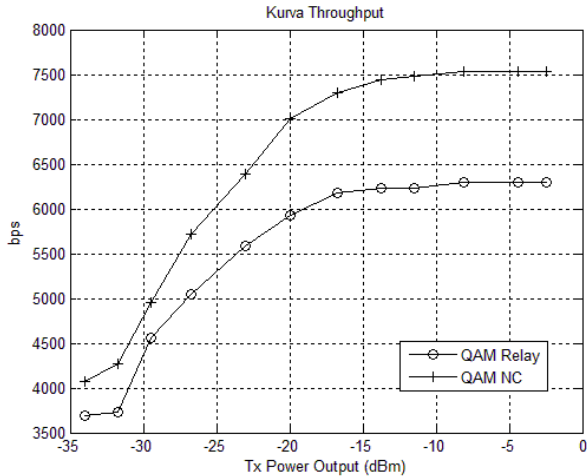


Hasil dari pengukuran perbandingan tiga sistem komunikasi, Sistem 16-QAM dengan *Network Coding*, 16-QAM tanpa *Network Coding*, dan 16-QAM *Point to Point* digambarkan pada gambar diatas. Garis kurva pada gambar diatas yang memiliki nilai BER paling rendah baik pada kondisi *indoor* maupun *outdoor* yaitu BER dari 16-QAM *point to point*. Hasil tersebut dikarenakan pada sistem *point to point* tidak menggunakan *relay*. Pada sistem *point to point* sinyal dipancarkan langsung dari sumber 1 ke sumber 2. Sehingga daya sinyal yang diterima lebih lemah dibandingkan dengan daya sinyal pada sistem yang menggunakan *relay*. Akan tetapi bila dibandingkan dengan hasil pada 4-QAM dan 16-QAM, hasil dari sistem *point to point* tidak memiliki selisih yang besar dengan kedua sistem lainnya.

Pada gambar diatas pada kondisi *indoor* dan *outdoor* 16-QAM dengan *network coding* memiliki nilai BER yang lebih rendah dari nilai BER 16-QAM tanpa *network coding*. Meskipun 16-QAM dengan *network coding* memiliki nilai BER yang lebih rendah namun perbedaannya dengan nilai BER dari 16-QAM tanpa *network coding* tidak signifikan dan nilai BER keduanya pun tetap tidak memberikan selisih BER yang besar sampai akhirnya BER bernilai 0. Dari hasil tersebut telah menggambarkan bahwa hasil kurva diatas mendekati kurva pada simulasi dan pada [7].



**Gambar 4.21** Kurva *Throughput* 16-QAM *Indoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*



**Gambar 4.22** Kurva *Throughput* 16-QAM *Outdoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

Gambar diatas merupakan hasil pengukuran *throughput* dari sistem 16-QAM dengan *network coding* dan sistem 16-QAM tanpa *network coding*. Hasil pengukuran pada kondisi *indoor* dan pada kondisi *outdoor* memiliki bentuk kurva yang hampir mirip, namun berbeda dari hasil pengukuran sebelumnya. Pada pengukuran dengan modulasi 16-QAM ini, nilai *throughput* pada kondisi *indoor* maupun *outdoor* memiliki perbedaan nilai yang tidak signifikan, tidak seperti 2 modulasi sebelumnya.

Pada kondisi lingkungan yang sama, baik pada kondisi *indoor* maupun pada kondisi *outdoor*, kurva dari 16-QAM dengan *network coding* menghasilkan nilai *throughput* yang jauh lebih baik dibandingkan dengan kurva dari 16-QAM tanpa *network coding*. Pada kondisi *indoor*, pada *power output* -11,5 dBm, nilai *throughput* dengan *network coding* yang dihasilkan mencapai 7,3966 kbps sedangkan tanpa *network coding* mencapai 6,2324 kbps. Bila dilakukan perhitungan dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

$$QAM\ NC - QAM\ Non\ NC = 7396.6 - 6232.4 = 1164.2$$

$$\frac{1164.2}{QAM\ Non\ NC} \times 100\% = 18.67\%$$

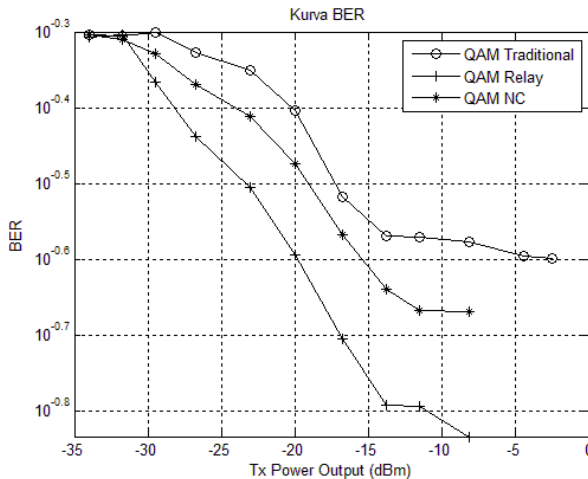
hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 18,67 %.

Dan bila ditinjau dari nilai maksimum *throughput* dari kedua sistem pada kondisi *outdoor*, pada *power output* -8,125 dBm nilai *throughput* dengan *network coding* mencapai 7,5332 kbps sedangkan tanpa *network coding* 6,2991 kbps. Bila dilakukan perhitungan persentase dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

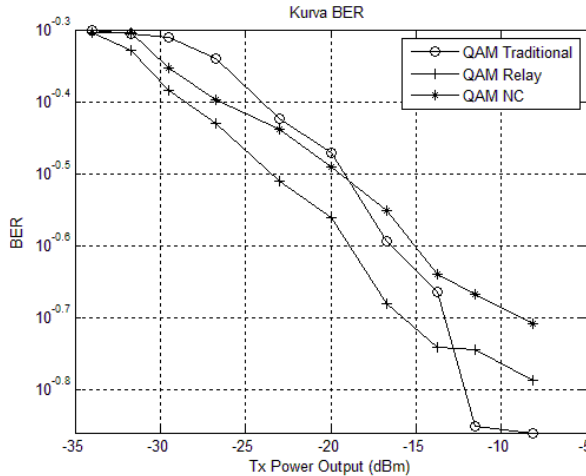
$$\frac{QAM\ NC - QAM\ Non\ NC}{1234.1} \times 100\% = 19.59\%$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 19.59 %.

#### 4.2.4 Analisis Sistem pada Modulasi 32-QAM



**Gambar 4.23** Kurva BER 32-QAM *Indoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

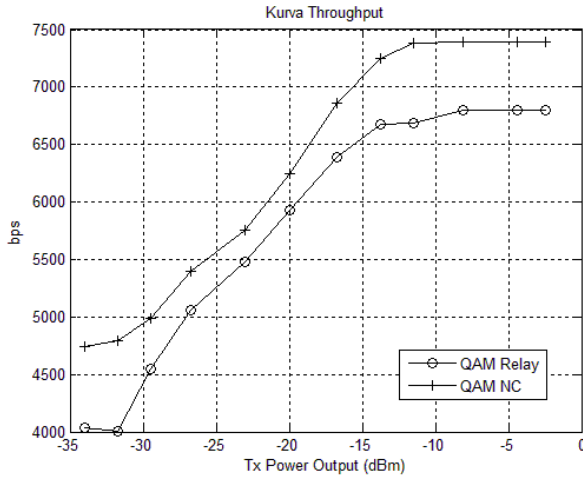


**Gambar 4.24** Kurva BER 32-QAM Outdoor dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

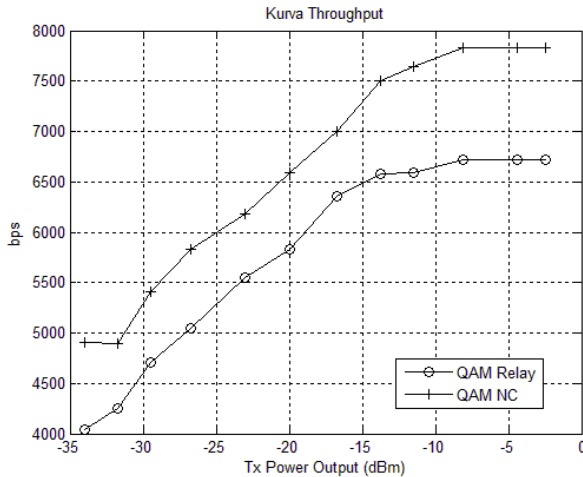
Hasil dari pengukuran perbandingan tiga sistem komunikasi, Sistem 32-QAM dengan *Network Coding*, 32-QAM tanpa *Network Coding*, dan 32-QAM *Point to point* digambarkan pada gambar diatas. Tidak seperti hasil pengukuran pada modulasi sebelum-sebelumnya, pada modulasi 32-QAM, hasil kurva yang dihasilkan dari ketiga sistem memiliki nilai selisih yang tidak besar. Selain itu pada kondisi lingkungan *outdoor*, kurva sistem *point to point* lebih bagus dari kedua sistem lainnya pada kondisi daya pancar maksimum.

Pada gambar diatas pada kondisi *indoor* dan *outdoor* 32-QAM dengan *network coding* memiliki nilai BER yang lebih rendah dari nilai BER 32-QAM tanpa *network coding*. Meskipun 32-QAM dengan *network coding* memiliki nilai BER yang lebih rendah namun perbedaannya dengan nilai BER dari 32-QAM tanpa *network coding* tidak signifikan dan nilai BER keduanya pun tetap tidak memberikan selisih BER yang besar sampai akhirnya BER bernilai 0. Dari hasil tersebut telah menggambarkan bahwa hasil kurva diatas mendekati kurva pada simulasi dan pada [7].





**Gambar 4.25** Kurva *Throughput* 32-QAM Indoor dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*



**Gambar 4.26** Kurva *Throughput* 32-QAM Outdoor dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

Hasil pengukuran *throughput* dari sistem 32-QAM dengan *network coding* dan sistem 32-QAM tanpa *network coding* ditampilkan pada gambar diatas. Kurva yang dihasilkan pada kondisi *indoor* dan pada kondisi *outdoor* memiliki bentuk kurva yang hampir mirip dan hasil dari dua kondisi lingkungan tersebut memiliki perbedaan nilai *throughput* yang tidak signifikan. Sama halnya dengan hasil pengukuran modulasi 16-QAM.

Pada kondisi lingkungan yang sama, baik pada kondisi *indoor* maupun pada kondisi *outdoor*, kurva dari 32-QAM dengan *network coding* masih menghasilkan nilai *throughput* yang jauh lebih baik dibandingkan dengan kurva dari 32-QAM tanpa *network coding*, sama seperti hasil modulasi sebelum-sebelumnya. Pada kondisi *indoor*, pada *power output* -8,125 dBm, nilai *throughput* dengan *network coding* yang dihasilkan mencapai 7,3938 kbps sedangkan tanpa *network coding* mencapai 6,682 kbps. Bila dilakukan perhitungan dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

$$\begin{aligned} QAM\ NC - QAM\ Non\ NC &= 7393.8 - 6682 = 711.8 \\ \frac{711.8}{QAM\ Non\ NC} \times 100\% &= 10.65\ \% \end{aligned}$$

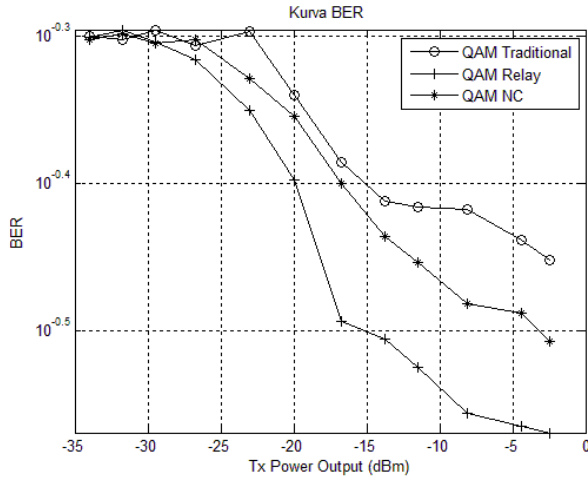
hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 10,65 %.

Dan bila ditinjau dari nilai maksimum *throughput* dari kedua sistem pada kondisi *outdoor*, pada *power output* -8,125 dBm nilai *throughput* dengan *network coding* mencapai 7,8354 kbps sedangkan tanpa *network coding* 6,7232 kbps. Bila dilakukan perhitungan persentase dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

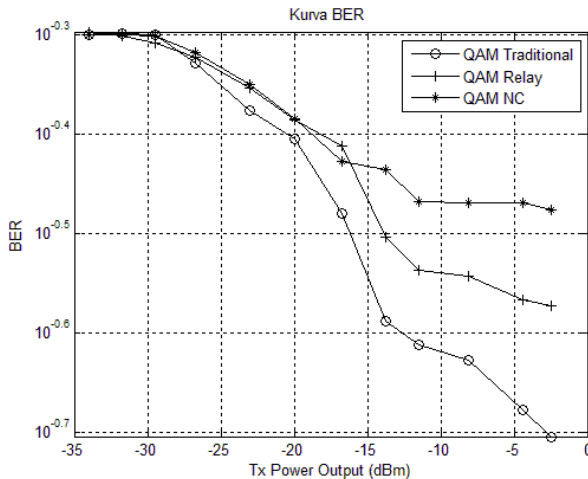
$$\begin{aligned} QAM\ NC - QAM\ Non\ NC &= 7835.4 - 6723.2 = 1112.2 \\ \frac{1112.2}{QAM\ Non\ NC} \times 100\% &= 16.54\ \% \end{aligned}$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 16,54 %.

#### 4.2.5 Analisis Sistem pada Modulasi 64-QAM



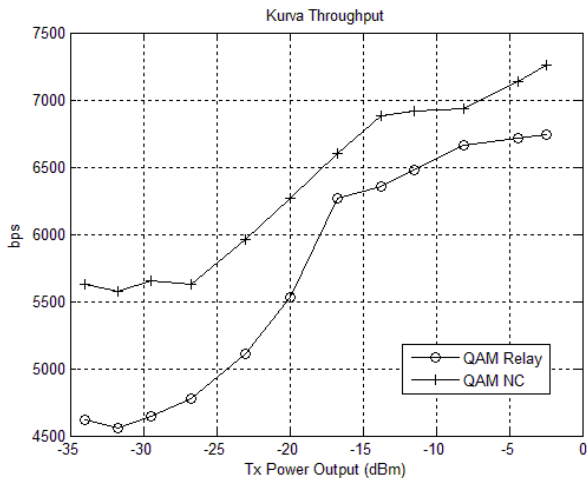
**Gambar 4.27** Kurva BER 64-QAM *Indoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*



**Gambar 4.28** Kurva BER 64-QAM *Outdoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

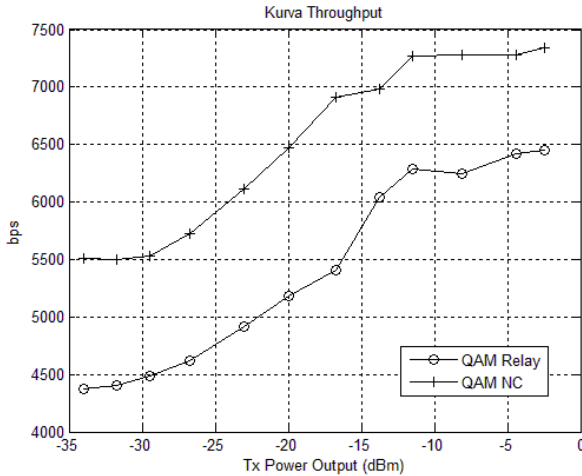
Hasil dari pengukuran perbandingan tiga sistem komunikasi, Sistem 64-QAM dengan *Network Coding*, 64-QAM tanpa *Network Coding*, dan 64-QAM *Point to point* digambarkan pada gambar diatas. Hasil kurva yang dihasilkan dari ketiga sistem memiliki nilai selisih yang tidak besar. Selain itu pada kondisi lingkungan *outdoor*, kurva sistem *point to point* lebih bagus dari kedua sistem lainnya dari daya pancar minimum hingga daya pancar maksimum.

Pada gambar diatas pada kondisi *indoor* dan *outdoor* 64-QAM dengan *network coding* memiliki nilai BER yang lebih rendah dari nilai BER 64-QAM tanpa *network coding*. Meskipun 64-QAM dengan *network coding* memiliki nilai BER yang lebih rendah namun perbedaannya dengan nilai BER dari 64-QAM tanpa *network coding* tidak signifikan dan nilai BER keduanya pun tetap tidak memberikan selisih BER yang besar sampai akhirnya BER bernilai 0. Dari hasil tersebut telah menggambarkan bahwa hasil kurva diatas mendekati kurva pada simulasi dan pada [7].



**Gambar 4.29** Kurva *Throughput* 64-QAM *Indoor* dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*





**Gambar 4.30** Kurva *Throughput* 64-QAM Indoor dari Sistem *Network Coding*, Tanpa *Network Coding*, dan *Point to Point*

Hasil pengukuran *throughput* dari sistem 64-QAM dengan *network coding* dan sistem 64-QAM tanpa *network coding* ditampilkan pada gambar diatas. Kurva yang dihasilkan pada kondisi *indoor* dan pada kondisi *outdoor* memiliki bentuk kurva yang hampir mirip dan hasil dari dua kondisi lingkungan tersebut memiliki perbedaan nilai *throughput* yang tidak signifikan. Sama halnya dengan hasil pengukuran modulasi 16-QAM dan 32-QAM.

Pada kondisi lingkungan yang sama, baik pada kondisi *indoor* maupun pada kondisi *outdoor*, kurva dari 64-QAM dengan *network coding* masih menghasilkan nilai *throughput* yang jauh lebih baik dibandingkan dengan kurva dari 64-QAM tanpa *network coding*, sama seperti hasil modulasi sebelum-sebelumnya. Pada kondisi *indoor*, pada *power output* -2,5 dBm, nilai *throughput* dengan *network coding* yang dihasilkan mencapai 7,2605 kbps sedangkan tanpa *network coding* mencapai 6,7433 kbps. Bila dilakukan perhitungan dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

$$QAM\ NC - QAM\ Non\ NC = 7260.5 - 6743.3 = 517.2$$

$$\frac{517.2}{QAM\ Non\ NC} \times 100\% = 7.66\%$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 7,66 %.

Dan bila ditinjau dari nilai maksimum *throughput* dari kedua sistem pada kondisi *outdoor*, pada *power output* -2,5 dBm nilai *throughput* dengan *network coding* mencapai 7,3442 kbps sedangkan tanpa *network coding* 6,450.1 kbps. Bila dilakukan perhitungan persentase dari selisih kedua nilai tersebut dijabarkan sebagai berikut:

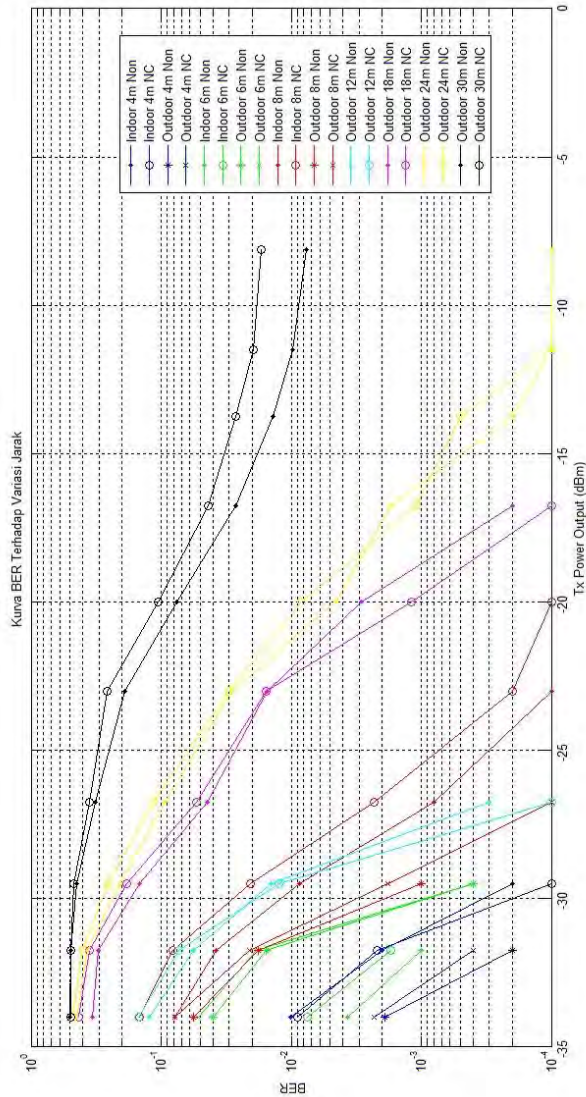
$$\begin{aligned} QAM\ NC - QAM\ Non\ NC &= 7344.2 - 6450.1 = 894.1 \\ \frac{894.1}{QAM\ Non\ NC} \times 100\% &= 13.86\% \end{aligned}$$

hasil dari perhitungan diatas sistem simulasi *network coding* mampu memperbaiki *throughput* hingga 13.86 %.

#### 4.3 Analisis Sistem *Network Coding* dengan Variasi Jarak

Pada pengukuran ini dilakukan pengamatan terhadap kinerja Sistem *Network Coding* terhadap variasi jarak. Secara teori, semakin besar jarak maka akan memberikan redaman yang besar pula terhadap sinyal yang dikirimkan. Pengukuran dilakukan pada kondisi lingkungan dalam ruangan maupun luar ruangan.

### 4.3.1 Analisis Sistem pada Modulasi 4-QAM



**Gambar 4.31** Kurva BER *Network Coding* 4-QAM Terhadap Variasi Jarak



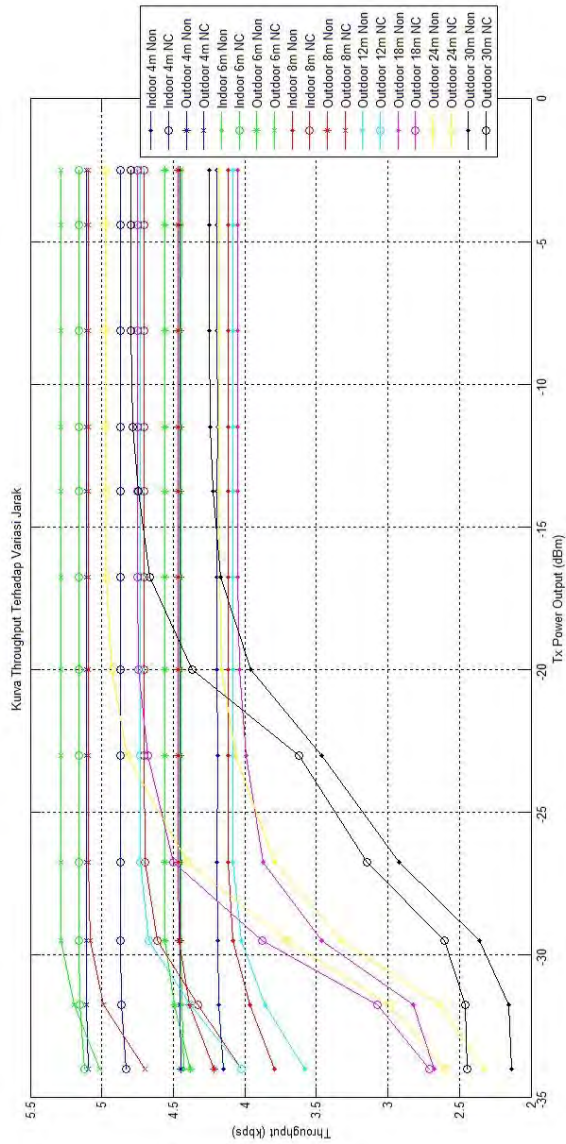
Pada Gambar 4.31 menggambarkan perbandingan kinerja 4-QAM dengan *network coding* dan 4-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Berdasarkan Gambar 4.31 jarak sangat mempengaruhi besar nilai BER dari 4-QAM dengan *network coding* dan 4-QAM tanpa *network coding*. Pada jarak terdekat, yaitu 4 meter, mempunyai nilai BER yang tinggi. Namun bila dibandingkan kembali dengan jarak yang sama, pengukuran yang dilakukan pada kondisi *outdoor* memiliki nilai BER yang lebih tinggi dibanding pada kondisi *indoor*.

Selanjutnya pada jarak 6 meter, pada kondisi *outdoor*, nilai BER yang dihasilkan lebih rendah dari nilai BER pada jarak 4 meter pada kondisi *outdoor* namun lebih tinggi dari nilai BER pada jarak 4 meter pada kondisi *indoor*. Untuk nilai BER pada kondisi *indoor* dengan jarak yang sama, memiliki nilai BER yang lebih rendah dari nilai BER jarak 4 meter pada kondisi *indoor*.

Hasil pengukuran selanjutnya terdapat nilai BER pada jarak 8 meter pada kondisi *outdoor* dan *indoor*. Dari kondisi jarak yang sama namun dengan kondisi lingkungan yang berbeda, nilai BER yang dihasilkan pada kondisi *outdoor* masih jauh lebih tinggi dibandingkan pada kondisi *indoor*.

Kemudian untuk nilai BER pada jarak 12 meter, 18 meter, 24 meter, dan 30 meter, memiliki nilai BER yang lebih rendah dibanding dengan nilai BER pada jarak-jarak sebelumnya. Posisi kurva dari tiap-tiap jarak berurutan dari jarak 12 meter hingga jarak 30 meter. Besar jarak antar *node* memberikan penurunan pada nilai BER. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai BER-nya.



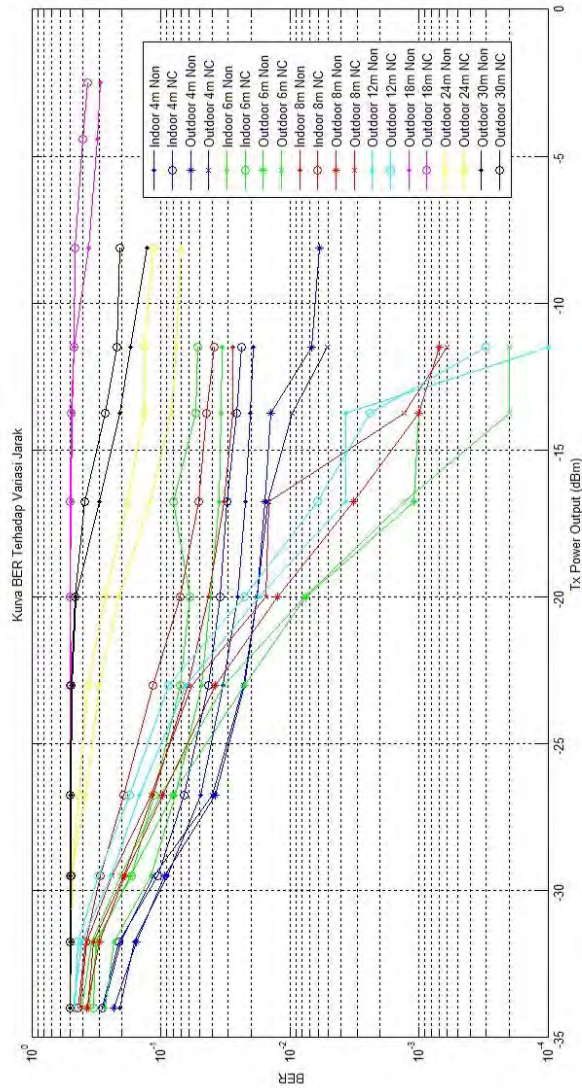


**Gambar 4.32** Kurva *Throughput Network Coding 4-QAM* Terhadap Variasi Jarak

Gambar 4.32 merupakan hasil pengukuran *throughput* dari sistem 4-QAM dengan *network coding* dan sistem 4-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Ditinjau dari hasil pengukuran pada setiap jarak, sistem dengan *network coding* selalu memiliki nilai *throughput* yang lebih baik dari sistem tanpa *network coding*. Di setiap jarak, pengukuran pada kondisi *outdoor* memiliki nilai *throughput* yang lebih tinggi dibandingkan pada kondisi *indoor*.

Pada jarak 6 meter, pada kondisi *outdoor*, sistem dengan *network coding* menghasilkan nilai *throughput* tertinggi dibanding dengan hasil pengukuran yang lainnya. Dibawah nilai *throughput* pada jarak 6 meter *outdoor*, terdapat kurva nilai *throughput* pada kondisi *indoor* pada jarak yang sama. Dan selanjutnya, terdapat kurva nilai *throughput* pada jarak 4 meter, pada nilai yang lebih rendah lagi terdapat kurva nilai *throughput* 8 meter, dan seterusnya berurutan hingga paling rendah dari 12 meter, 18 meter, 24 meter, hingga 30 meter. Besar jarak antar *node* memberikan penurunan pada nilai *throughput*. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai *throughput*-nya. Selain itu, dengan daya pancar yang rendah, memberikan efek yang berbeda-beda pada bentuk awal kurva di setiap variasi jarak antar *node*. Semakin besar jaraknya pada daya pancar yang rendah, nilai *throughput* yang dihasilkan semakin rendah dan semakin jauh dari nilai maksimum *throughput*-nya.

### 4.3.2 Analisis Sistem pada Modulasi 8-QAM



**Gambar 4.33** Kurva BER *Network Coding* 8-QAM Terhadap Variasi Jarak



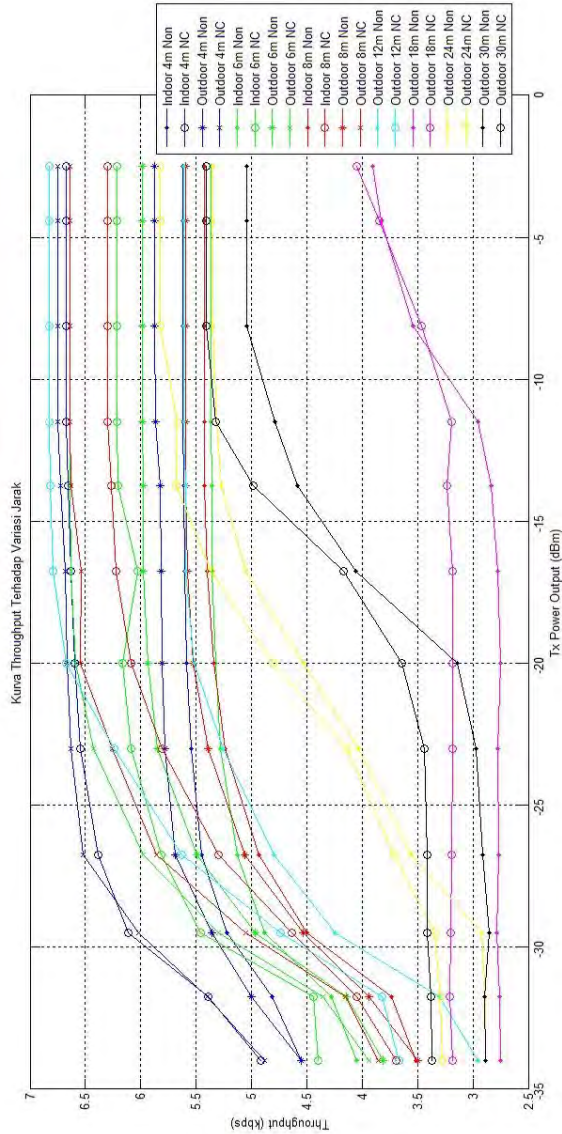
Pada Gambar 4.33 menggambarkan perbandingan kinerja 8-QAM dengan *network coding* dan 8-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Berdasarkan Gambar 4.33 jarak sangat mempengaruhi besar nilai BER dari 8-QAM dengan *network coding* dan 8-QAM tanpa *network coding*. Pada jarak terdekat, yaitu 4 meter, mempunyai nilai BER yang tinggi. Namun bila dibandingkan kembali dengan jarak yang sama, pengukuran yang dilakukan pada kondisi *outdoor* memiliki nilai BER yang lebih tinggi dibanding pada kondisi *indoor*. Hal tersebut juga terjadi pada jarak-jarak yang lainnya.

Selanjutnya pada jarak-jarak yang lebih besar menghasilkan nilai BER yang lebih rendah pula dan terus berurut dari jarak yang terkecil hingga jarak yang terbesar. Besar jarak antar *node* memberikan penurunan pada nilai BER. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai BER-nya.

Selain itu, semakin besar jarak juga mempengaruhi peningkatan nilai BER. Pada jarak 6 meter, 8 meter, dan 12 meter dilihat pada kurva nilai BER mengalami peningkatan yang signifikan. Kurva terlihat seperti membentuk sebuah lembah. Tetapi pada jarak yang lainnya, dilihat pada kurva nilai BER mengalami peningkatan yang tidak signifikan, kurva terlihat landai.

Lalu jika dibandingkan kembali dengan Gambar 4.31, kurva yang dihasilkan pada sistem modulasi 8-QAM di setiap variasi jarak memiliki peningkatan nilai BER yang tidak signifikan. Dan bila dilihat dari bentuknya, kurva pada modulasi 8-QAM memberikan bentuk yang landai.





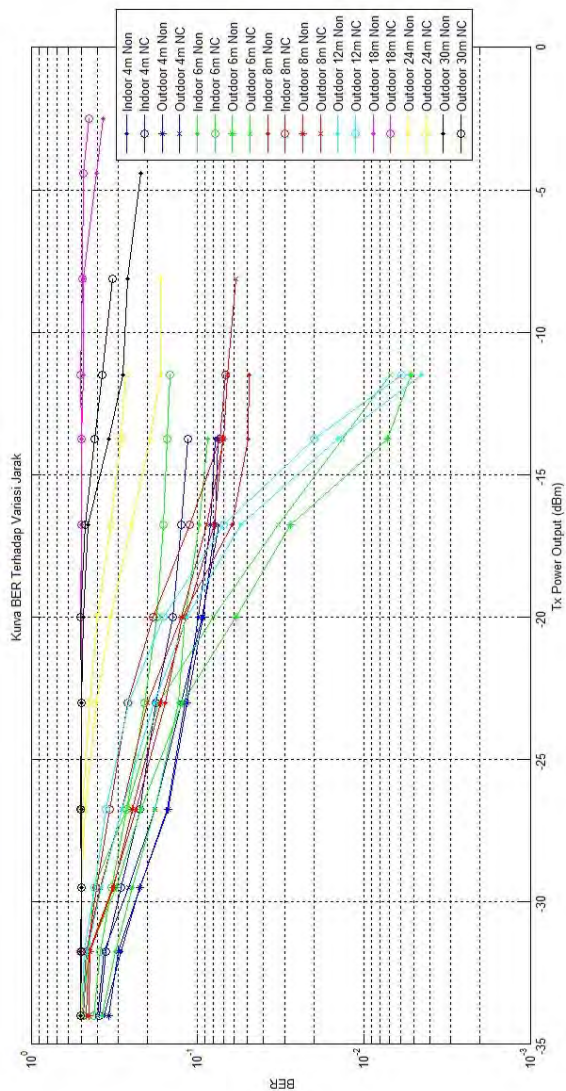
**Gambar 4.34** Kurva *Throughput Network Coding 8-QAM* Terhadap Variasi Jarak

Gambar 4.34 merupakan hasil pengukuran *throughput* dari sistem 8-QAM dengan *network coding* dan sistem 8-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Ditinjau dari hasil pengukuran pada setiap jarak, sistem dengan *network coding* selalu memiliki nilai *throughput* yang lebih baik dari sistem tanpa *network coding*. Di setiap jarak, pengukuran pada kondisi *outdoor* memiliki nilai *throughput* yang lebih tinggi dibandingkan pada kondisi *indoor*.

Pada modulasi 8-QAM, jarak 12 meter pada kondisi *outdoor*, sistem dengan *network coding* menghasilkan nilai *throughput* tertinggi dibanding dengan hasil pengukuran yang lainnya. Meskipun di awal kurva yang memiliki nilai *throughput* tertinggi yaitu pada jarak 4 meter pada kondisi *outdoor*.

Pada nilai yang lebih rendah lagi terdapat kurva nilai *throughput* 8 meter, dan seterusnya berurutan hingga paling rendah dari 12 meter, 18 meter, 24 meter, hingga 30 meter. Besar jarak antar *node* memberikan penurunan pada nilai *throughput*. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai *throughput*-nya. Selain itu, dengan daya pancar yang rendah, memberikan efek yang berbeda-beda pada bentuk awal kurva di setiap variasi jarak antar *node*. Semakin besar jaraknya pada daya pancar yang rendah, nilai *throughput* yang dihasilkan semakin rendah dan semakin jauh dari nilai maksimum *throughput*-nya.

### 4.3.3 Analisis Sistem pada Modulasi 16-QAM



**Gambar 4.35** Kurva BER *Network Coding* 16-QAM Terhadap Variasi Jarak



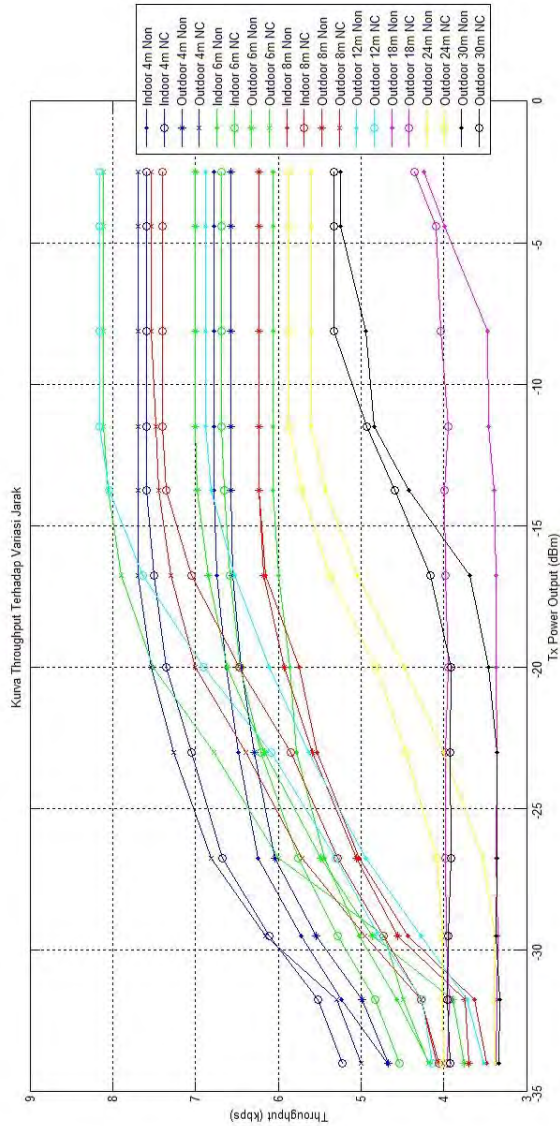
Pada Gambar 4.35 menggambarkan perbandingan kinerja 16-QAM dengan *network coding* dan 16-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Berdasarkan Gambar 4.35 diatas jarak sangat mempengaruhi besar nilai BER dari 16-QAM dengan *network coding* dan 16-QAM tanpa *network coding*. Pada jarak terdekat, yaitu 4 meter, mempunyai nilai BER yang tertinggi. Namun bila dibandingkan kembali dengan jarak yang sama, pengukuran yang dilakukan pada kondisi *outdoor* memiliki nilai BER yang lebih tinggi dibanding pada kondisi *indoor*. Hal tersebut juga terjadi pada jarak-jarak yang lainnya.

Selanjutnya pada jarak-jarak yang lebih besar menghasilkan nilai BER yang lebih rendah pula dan terus berurut dari jarak yang terkecil hingga jarak yang terbesar. Besar jarak antar *node* memberikan penurunan pada nilai BER. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai BER-nya.

Selain itu, semakin besar jarak juga mempengaruhi peningkatan nilai BER. Pada jarak 6 meter dan 12 meter, dilihat pada kurva nilai BER mengalami peningkatan yang signifikan. Kurva terlihat seperti membentuk sebuah lembah. Tetapi pada jarak lainnya, dilihat pada kurva nilai BER mengalami peningkatan yang tidak signifikan, kurva terlihat landai.

Lalu jika dibandingkan kembali dengan Gambar 4.31 dan Gambar 4.33, kurva yang dihasilkan pada sistem modulasi 16-QAM di setiap variasi jarak memiliki peningkatan nilai BER yang lebih tidak signifikan. Dan bila dilihat dari bentuknya, kurva pada modulasi 16-QAM memberikan bentuk yang lebih landai.





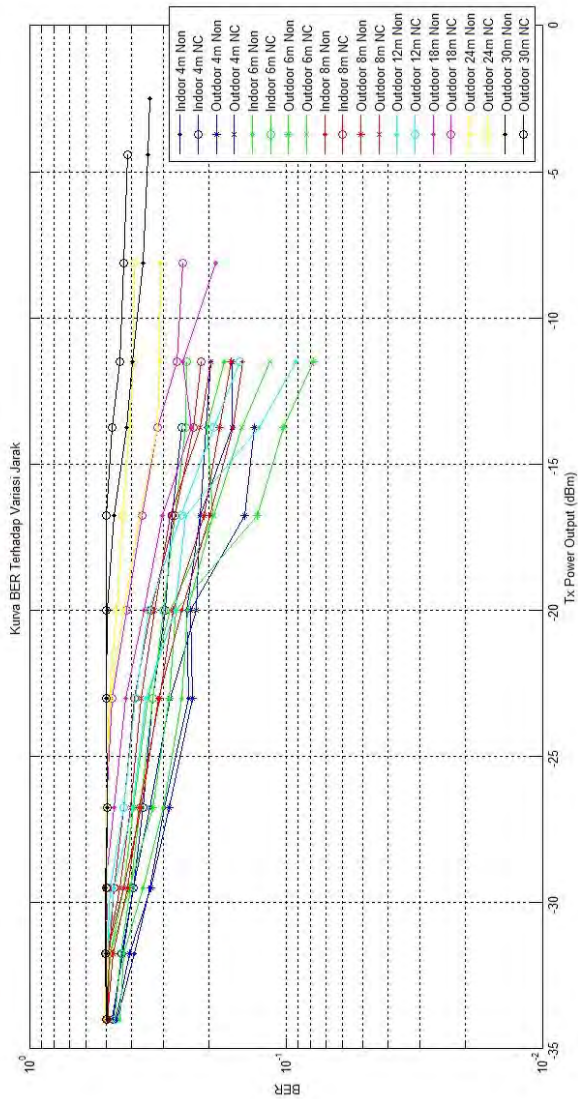
**Gambar 4.36** Kurva *Throughput Network Coding 16-QAM* Terhadap Variasi Jarak

Gambar 4.36 merupakan hasil pengukuran *throughput* dari sistem 16-QAM dengan *network coding* dan sistem 16-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Ditinjau dari hasil pengukuran pada setiap jarak, sistem dengan *network coding* selalu memiliki nilai *throughput* yang lebih baik dari sistem tanpa *network coding*. Di setiap jarak, pengukuran pada kondisi *outdoor* memiliki nilai *throughput* yang lebih tinggi dibandingkan pada kondisi *indoor*.

Pada modulasi 16-QAM, jarak 12 meter pada kondisi *outdoor*, sistem dengan *network coding* menghasilkan nilai *throughput* tertinggi dibanding dengan hasil pengukuran yang lainnya. Meskipun di awal kurva yang memiliki nilai *throughput* tertinggi yaitu pada jarak 4 meter pada kondisi *outdoor*.

Pada nilai yang lebih rendah lagi terdapat kurva nilai *throughput* 8 meter, dan seterusnya berurutan hingga paling rendah dari 12 meter, 18 meter, 24 meter, hingga 30 meter. Besar jarak antar *node* memberikan penurunan pada nilai *throughput*. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai *throughput*-nya. Selain itu, dengan daya pancar yang rendah, memberikan efek yang berbeda-beda pada bentuk awal kurva di setiap variasi jarak antar *node*. Semakin besar jaraknya pada daya pancar yang rendah, nilai *throughput* yang dihasilkan semakin rendah dan semakin jauh dari nilai maksimum *throughput*-nya.

#### 4.3.4 Analisis Sistem pada Modulasi 32-QAM



**Gambar 4.37** Kurva BER *Network Coding* 32-QAM Terhadap Variasi Jarak

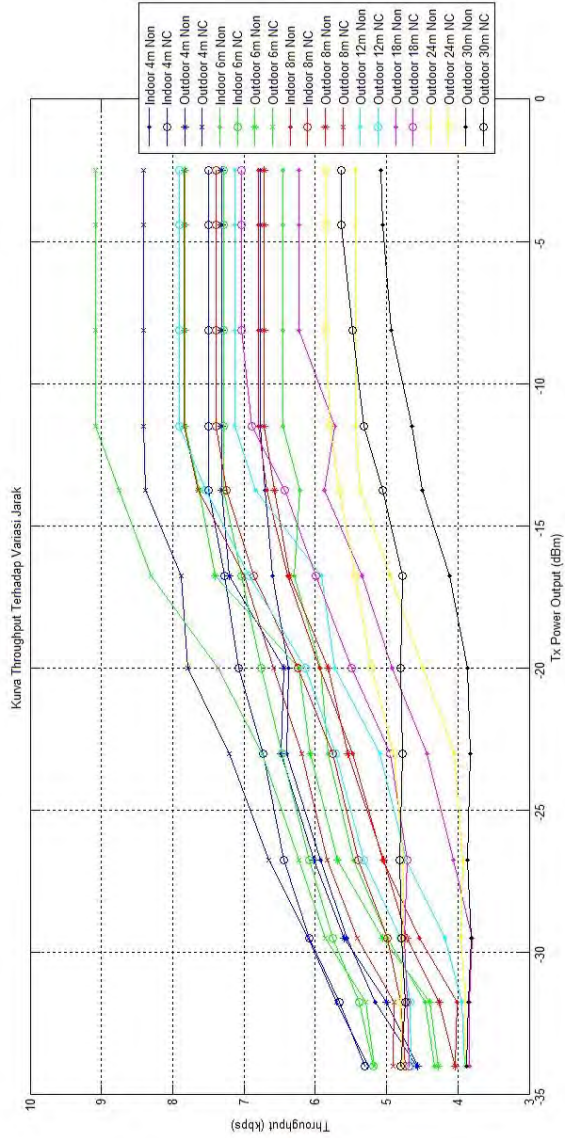
Pada Gambar 4.37 menggambarkan perbandingan kinerja 32-QAM dengan *network coding* dan 32-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Berdasarkan Gambar 4.37 jarak sangat mempengaruhi besar nilai BER dari 32-QAM dengan *network coding* dan 32-QAM tanpa *network coding*. Pada jarak terdekat, yaitu 4 meter, mempunyai nilai BER yang tertinggi. Namun bila dibandingkan kembali dengan jarak yang sama, pengukuran yang dilakukan pada kondisi *outdoor* memiliki nilai BER yang lebih tinggi dibanding pada kondisi *indoor*. Hal tersebut juga terjadi pada jarak-jarak yang lainnya.

Selanjutnya pada jarak-jarak yang lebih besar menghasilkan nilai BER yang lebih rendah pula dan terus berurut dari jarak yang terkecil hingga jarak yang terbesar. Besar jarak antar *node* memberikan penurunan pada nilai BER. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai BER-nya.

Selain itu, semakin besar jarak juga mempengaruhi peningkatan nilai BER. Pada jarak 6 meter dan 12 meter, dilihat pada kurva nilai BER mengalami peningkatan yang signifikan. Kurva terlihat seperti membentuk sebuah lembah. Tetapi pada jarak lainnya, dilihat pada kurva nilai BER mengalami peningkatan yang tidak signifikan, kurva terlihat landai.

Lalu jika dibandingkan kembali dengan Gambar 3.1, Gambar 3.3 dan Gambar 3.5, kurva yang dihasilkan pada sistem modulasi 32-QAM di setiap variasi jarak memiliki peningkatan nilai BER yang lebih tidak signifikan. Dan bila dilihat dari bentuknya, kurva pada modulasi 32-QAM memberikan bentuk yang lebih landai.





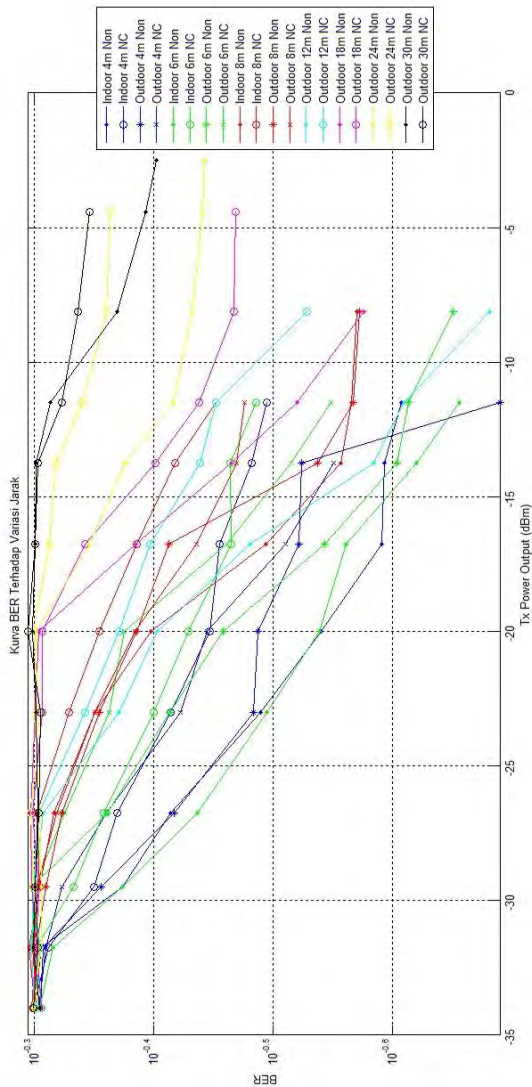
**Gambar 4.38** Kurva *Throughput Network Coding 32-QAM* Terhadap Variasi Jarak

Gambar 4.38 merupakan hasil pengukuran *throughput* dari sistem 16-QAM dengan *network coding* dan sistem 32-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Ditinjau dari hasil pengukuran pada setiap jarak, sistem dengan *network coding* selalu memiliki nilai *throughput* yang lebih baik dari sistem tanpa *network coding*. Di setiap jarak, pengukuran pada kondisi *outdoor* memiliki nilai *throughput* yang lebih tinggi dibandingkan pada kondisi *indoor*.

Pada modulasi 32-QAM, jarak 6 meter pada kondisi *outdoor*, sistem dengan *network coding* menghasilkan nilai *throughput* tertinggi dibanding dengan hasil pengukuran yang lainnya. Meskipun di awal kurva yang memiliki nilai *throughput* tertinggi yaitu pada jarak 4 meter pada kondisi *outdoor*.

Pada nilai yang lebih rendah lagi terdapat kurva nilai *throughput* 8 meter, dan seterusnya berurutan hingga paling rendah dari 12 meter, 18 meter, 24 meter, hingga 30 meter. Besar jarak antar *node* memberikan penurunan pada nilai *throughput*. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai *throughput*-nya. Selain itu, dengan daya pancar yang rendah, memberikan efek yang berbeda-beda pada bentuk awal kurva di setiap variasi jarak antar *node*. Semakin besar jaraknya pada daya pancar yang rendah, nilai *throughput* yang dihasilkan semakin rendah dan semakin jauh dari nilai maksimum *throughput*-nya.

### 4.3.5 Analisis Sistem pada Modulasi 64-QAM



**Gambar 4.39** Kurva BER *Network Coding* 64-QAM Terhadap Variasi Jarak

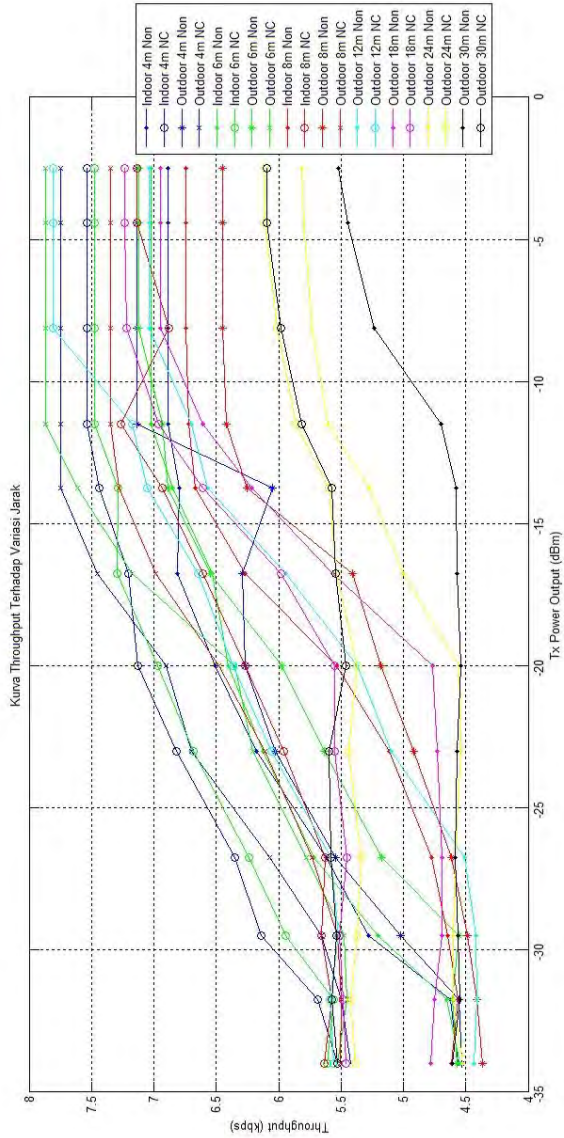
Pada Gambar 4.39 menggambarkan perbandingan kinerja 64-QAM dengan *network coding* dan 64-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Berdasarkan Gambar 4.39 jarak sangat mempengaruhi besar nilai BER dari 64-QAM dengan *network coding* dan 64-QAM tanpa *network coding*. Pada jarak terdekat, yaitu 4 meter, mempunyai nilai BER yang tertinggi. Namun bila dibandingkan kembali dengan jarak yang sama, pengukuran yang dilakukan pada kondisi *outdoor* memiliki nilai BER yang lebih tinggi dibanding pada kondisi *indoor*. Hal tersebut juga terjadi pada jarak-jarak yang lainnya.

Selanjutnya pada jarak-jarak yang lebih besar menghasilkan nilai BER yang lebih rendah pula dan terus berurut dari jarak yang terkecil hingga jarak yang terbesar. Besar jarak antar *node* memberikan penurunan pada nilai BER. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai BER-nya.

Pada hasil pengukuran modulasi 64-QAM berbeda dengan hasil pengukuran modulasi 8-QAM, 16-QAM, dan 32-QAM, kurva nilai BER yang dihasilkan tidak berbentuk landai seperti sebelumnya. Hanya saja nilai BER yang dihasilkan lebih rendah dibanding dengan modulasi sebelum-sebelumnya.

Lalu jika dibandingkan kembali dengan Gambar 3.3, Gambar 3.5, dan Gambar 3.7, kurva yang dihasilkan pada sistem modulasi 64-QAM di setiap variasi jarak memiliki peningkatan nilai BER yang lebih signifikan. Dan bila dilihat dari bentuknya, kurva pada modulasi 64-QAM memberikan bentuk yang tidak landai.





**Gambar 4.40** Kurva *Throughput Network Coding 64-QAM* Terhadap Variasi Jarak

Gambar 4.40 merupakan hasil pengukuran *throughput* dari sistem 16-QAM dengan *network coding* dan sistem 64-QAM tanpa *network coding* di setiap jarak pada kondisi *indoor* dan kondisi *outdoor*. Ditinjau dari hasil pengukuran pada setiap jarak, sistem dengan *network coding* selalu memiliki nilai *throughput* yang lebih baik dari sistem tanpa *network coding*. Di setiap jarak, pengukuran pada kondisi *outdoor* memiliki nilai *throughput* yang lebih tinggi dibandingkan pada kondisi *indoor*.

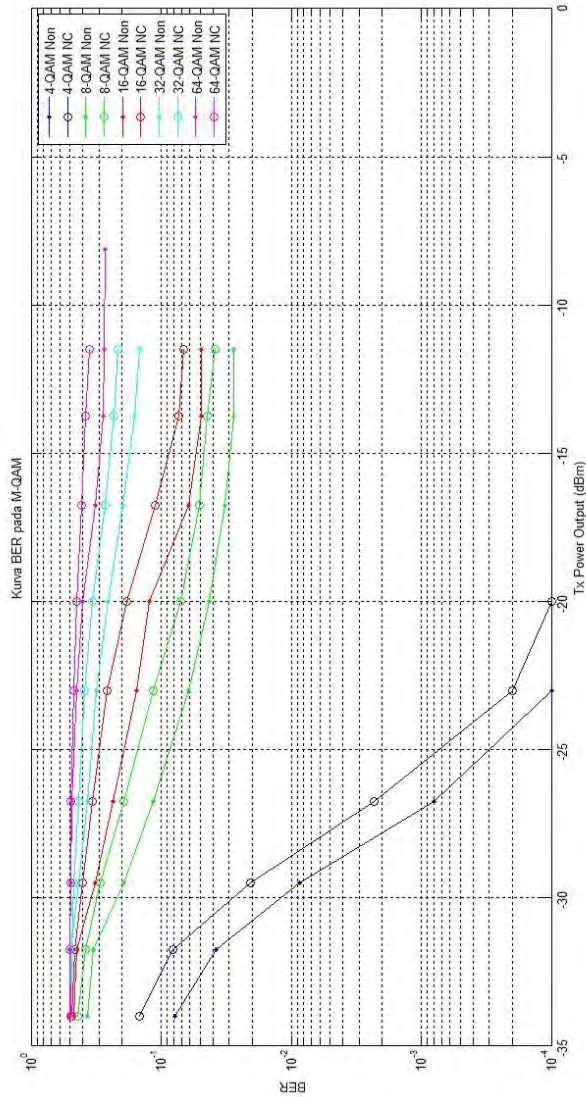
Pada modulasi 64-QAM, jarak 6 meter pada kondisi *outdoor*, sistem dengan *network coding* menghasilkan nilai *throughput* tertinggi dibanding dengan hasil pengukuran yang lainnya. Meskipun di awal kurva yang memiliki nilai *throughput* tertinggi yaitu pada jarak 4 meter pada kondisi *outdoor*.

Pada nilai yang lebih rendah lagi terdapat kurva nilai *throughput* 8 meter, dan seterusnya berurutan hingga paling rendah dari 12 meter, 18 meter, 24 meter, hingga 30 meter. Besar jarak antar *node* memberikan penurunan pada nilai *throughput*. Semakin besar jarak antar *node* maka semakin besar penurunan pada nilai *throughput*-nya. Selain itu, dengan daya pancar yang rendah, memberikan efek yang berbeda-beda pada bentuk awal kurva di setiap variasi jarak antar *node*. Semakin besar jaraknya pada daya pancar yang rendah, nilai *throughput* yang dihasilkan semakin rendah dan semakin jauh dari nilai maksimum *throughput*-nya.

#### **4.4 Analisis Sistem *Network Coding* pada Modulasi M-QAM**

Pada pengukuran ini dilakukan pengamatan terhadap Sistem *Network Coding* pada Modulasi M-QAM. Pengamatan dilakukan dengan membandingkan kinerja dari *network coding* di setiap modulasi QAM. Kinerja sistem diamati dari dua parameter, yaitu BER dan *Throughput*. Pengukuran dilakukan pada kondisi lingkungan *Indoor* dan lingkungan *Outdoor*. Hasil dari pengukuran yang digunakan sebagai analisis merupakan hasil pengukuran pada jarak 8 meter.

#### 4.4.1 Analisis Berdasarkan *Bit Error Rate*



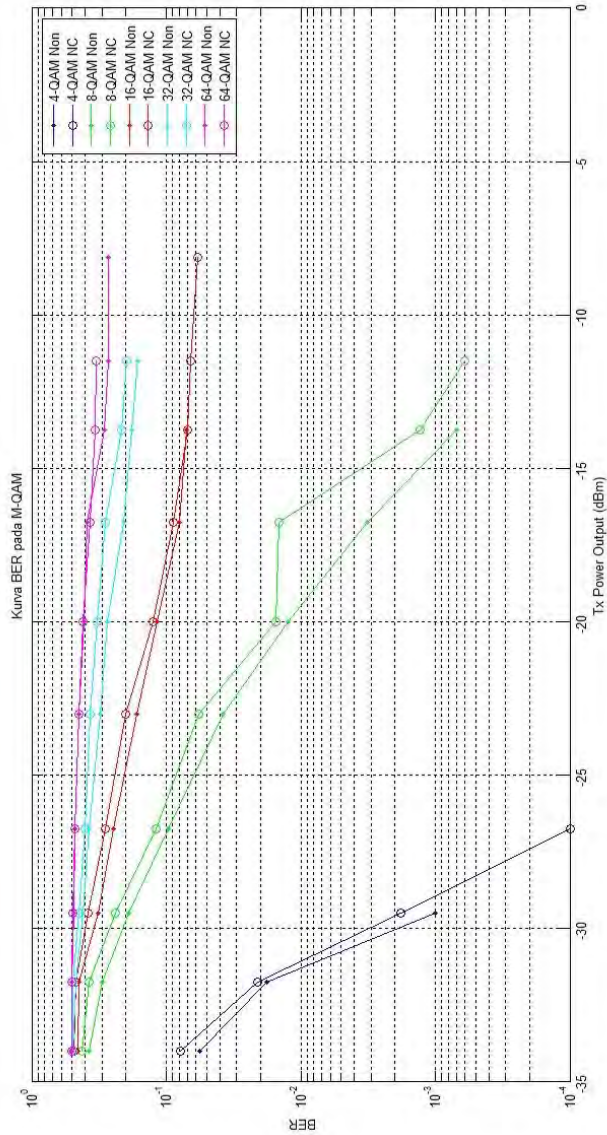
**Gambar 4.41** Kurva BER *Network Coding Indoor* pada Modulasi M-QAM



Gambar 4.41 menggambarkan perbandingan kurva BER sistem dengan *network coding* dan sistem tanpa *network coding* pada setiap modulasi M-QAM pada kondisi *indoor*. Dari hasil pengukuran diatas terlihat level konstelasi mempengaruhi nilai BER baik pada sistem yang menggunakan *network coding* maupun sistem yang tidak menggunakan *network coding*. Modulasi 4-QAM menghasilkan nilai BER yang lebih tinggi dari level konstelasi yang lainnya. Berikutnya terdapat kurva BER dari modulasi 8-QAM yang lebih rendah dibandingkan dengan nilai BER dari modulasi 4-QAM. Selanjutnya merupakan kurva BER dari modulasi 16-QAM yang memiliki nilai BER lebih rendah dari nilai BER dari modulasi 4-QAM dan 8-QAM. Lalu untuk modulasi 32-QAM menghasilkan nilai BER yang lebih rendah lagi dari modulasi 16-QAM. Sampai pada hasil dimana modulasi 64-QAM memiliki nilai BER yang terendah dibandingkan dengan level konstelasi lainnya. Kelima modulasi tersebut mempunyai nilai BER yang berurut tertinggi hingga terendah dari 4-QAM, 8-QAM, 16-QAM, 32-QAM, hingga 64-QAM.

Bentuk kurva yang dihasilkan oleh modulasi 4-QAM memberikan bentuk kurva seperti lembah. Bentuk kurva seperti ini menjelaskan bahwa perubahan nilai BER yang terus meningkat terjadi secara signifikan. Sedangkan untuk modulasi 8-QAM, 16-QAM, 32-QAM, dan 64-QAM, keempat modulasi tersebut memberikan bentuk kurva yang landai. Semakin tinggi level konstelasi yang digunakan maka semakin landai pula bentuk kurva yang dihasilkan. Bentuk kurva yang landai tersebut menjelaskan bahwa perubahan nilai BER yang terus meningkat terjadi tidak secara signifikan. Karena perubahan yang tidak signifikan itulah yang menyebabkan bentuk kurva menjadi landai.





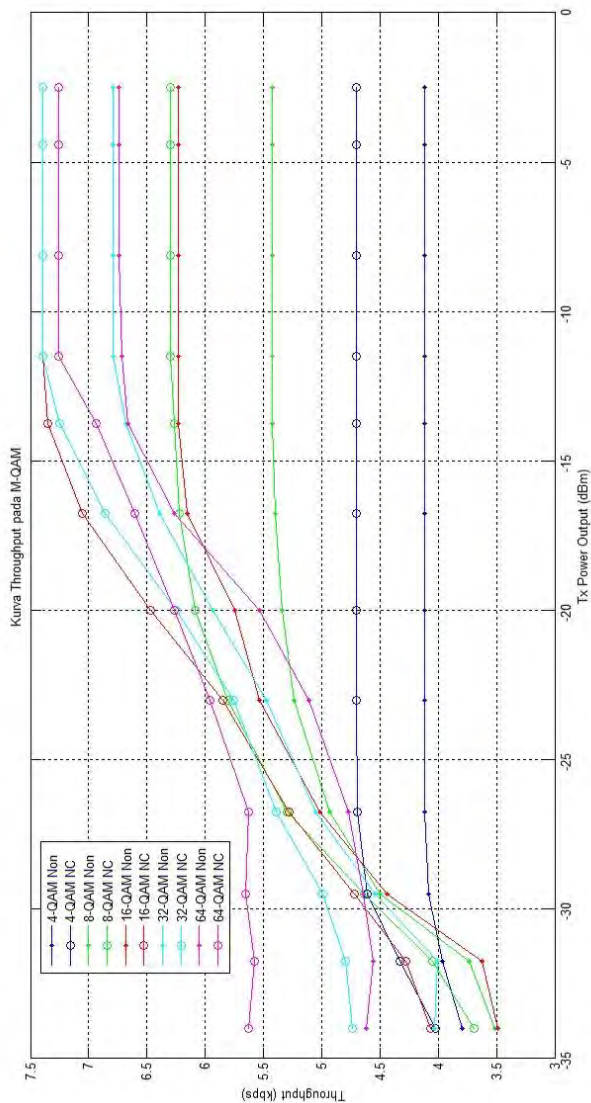
**Gambar 4.42** Kurva BER *Network Coding Outdoor* pada Modulasi M-QAM

Gambar 4.42 menggambarkan perbandingan kurva BER sistem dengan *network coding* dan sistem tanpa *network coding* pada setiap modulasi M-QAM pada kondisi *outdoor*. Dari hasil pengukuran diatas tidak jauh berbeda dengan hasil pengukuran pada kondisi *indoor*. Akan tetapi nilai BER yang dihasilkan pada pengukuran pada kondisi *outdoor* ini lebih tinggi dibandingkan dengan hasil pengukuran pada kondisi *indoor*.

Modulasi 4-QAM menghasilkan nilai BER yang lebih tinggi dari level konstelasi yang lainnya. Berikutnya terdapat kurva BER dari modulasi 8-QAM yang lebih rendah dibandingkan dengan nilai BER dari modulasi 4-QAM. Selanjutnya merupakan kurva BER dari modulasi 16-QAM yang memiliki nilai BER lebih rendah dari nilai BER dari modulasi 4-QAM dan 8-QAM. Lalu untuk modulasi 32-QAM menghasilkan nilai BER yang lebih rendah lagi dari modulasi 16-QAM. Sampai pada hasil dimana modulasi 64-QAM memiliki nilai BER yang terendah dibandingkan dengan level konstelasi lainnya. Kelima modulasi tersebut mempunyai nilai BER yang berurut tertinggi hingga terendah dari 4-QAM, 8-QAM, 16-QAM, 32-QAM, hingga 64-QAM.

Bentuk kurva yang dihasilkan oleh modulasi 4-QAM memberikan bentuk kurva yang lurus terjal ke bawah. Bentuk kurva yang dimaksud bahwa perubahan nilai BER yang terus meningkat terjadi secara signifikan pada setiap perubahan daya pancar pengirim meningkat. Berbeda dengan bentuk kurva yang dihasilkan pada kondisi *indoor*, pada kondisi *outdoor* bentuk kurva yang dihasilkan modulasi 8-QAM tidak lagi landai seperti bentuk kurva 8-QAM pada kondisi *indoor*. Sedangkan untuk modulasi 16-QAM, 32-QAM, dan 64-QAM, ketiga modulasi tersebut memberikan bentuk kurva yang sama seperti pada kondisi *indoor* yaitu memiliki bentuk kurva yang landai. Semakin tinggi level konstelasi yang digunakan maka semakin landai pula bentuk kurva yang dihasilkan. Bentuk kurva yang landai tersebut menjelaskan bahwa perubahan nilai BER yang terus meningkat terjadi tidak secara signifikan. Karena perubahan yang tidak signifikan itulah yang menyebabkan bentuk kurva menjadi landai.

#### 4.4.2 Analisis Berdasarkan *Throughput*

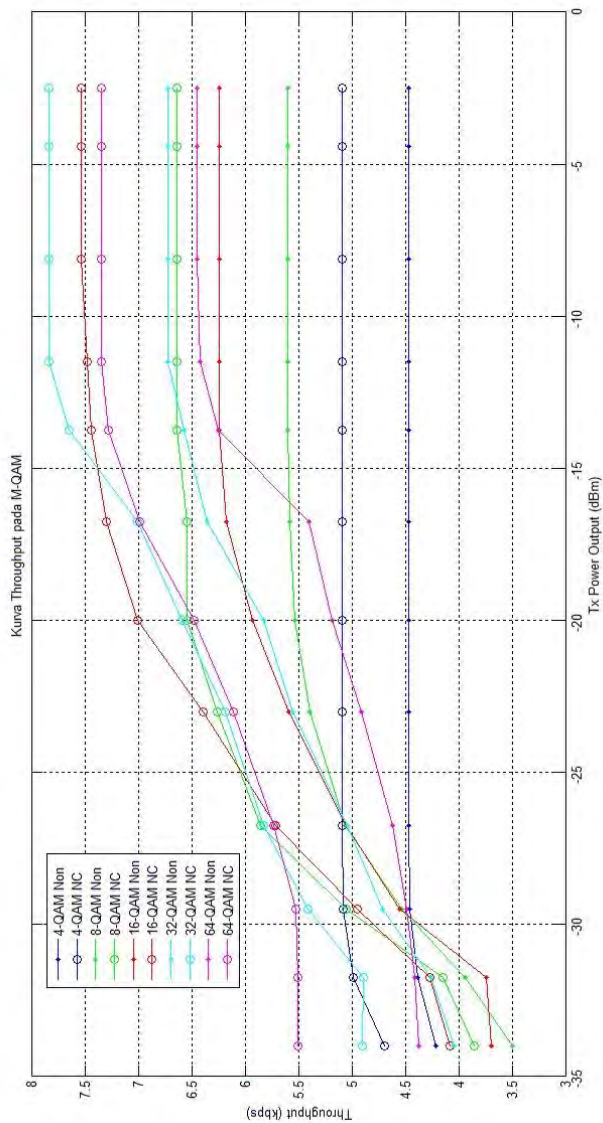


**Gambar 4.43** Kurva *Throughput Network Coding Indoor* pada Modulasi M-QAM



Pada gambar 4.43 menggambarkan perbandingan kurva *throughput* sistem dengan *network coding* dan sistem tanpa *network coding* pada setiap modulasi M-QAM pada kondisi *indoor*. Dari hasil pengukuran diatas terlihat level konstelasi mempengaruhi nilai *throughput* baik pada sistem yang menggunakan *network coding* maupun sistem yang tidak menggunakan *network coding*. Modulasi 4-QAM memiliki nilai *throughput* terendah dibandingkan dengan nilai *throughput* yang dimiliki level konstelasi yang lainnya. Berikutnya terdapat kurva *throughput* dari modulasi 8-QAM yang lebih tinggi dibandingkan dengan nilai *throughput* dari modulasi 4-QAM. Selanjutnya merupakan kurva *throughput* dari modulasi 16-QAM yang memiliki nilai *throughput* lebih tinggi dari nilai *throughput* dari modulasi 4-QAM dan 8-QAM. Lalu untuk modulasi 32-QAM menghasilkan nilai *throughput* yang lebih tinggi lagi dari modulasi 16-QAM. Sampai pada hasil dimana modulasi 64-QAM memiliki nilai *throughput* yang lebih tinggi lagi dibandingkan dengan level konstelasi lainnya. Akan tetapi nilai *throughput* dari modulasi 16-QAM dengan *network coding* dan dari modulasi 32-QAM dengan *network coding* melampaui nilai *throughput* dari modulasi 64-QAM dengan *network coding*. Dari hasil perbandingan tersebut, terlihat jelas bahwa *network coding* pada modulasi 16-QAM memberikan perbaikan *throughput* yang lebih besar dibandingkan pada level konstelasi lainnya.





**Gambar 4.44** Kurva *Throughput Network Coding Outdoor* pada Modulasi M-QAM

Pada gambar 4.44 menggambarkan perbandingan kurva *throughput* sistem dengan *network coding* dan sistem tanpa *network coding* pada setiap modulasi M-QAM pada kondisi *outdoor*. Dari hasil pengukuran diatas terlihat level konstelasi mempengaruhi nilai *throughput* baik pada sistem yang menggunakan *network coding* maupun sistem yang tidak menggunakan *network coding*. Modulasi 4-QAM memiliki nilai *throughput* terendah dibandingkan dengan nilai *throughput* yang dimiliki level konstelasi yang lainnya. Berikutnya terdapat kurva *throughput* dari modulasi 8-QAM yang lebih tinggi dibandingkan dengan nilai *throughput* dari modulasi 4-QAM. Selanjutnya merupakan kurva *throughput* dari modulasi 16-QAM yang memiliki nilai *throughput* lebih tinggi dari nilai *throughput* dari modulasi 4-QAM dan 8-QAM. Lalu untuk modulasi 32-QAM menghasilkan nilai *throughput* yang lebih tinggi lagi dari modulasi 16-QAM. Sampai pada hasil dimana modulasi 64-QAM memiliki nilai *throughput* yang lebih tinggi lagi dibandingkan dengan level konstelasi lainnya. Akan tetapi nilai *throughput* dari modulasi 16-QAM dengan *network coding* dan dari modulasi 32-QAM dengan *network coding* melampaui nilai *throughput* dari modulasi 64-QAM dengan *network coding* sama seperti hasil pada kondisi *indoor*.

Perbedaan dengan hasil pada kondisi *indoor*, hasil dari modulasi 32-QAM dengan *network coding* memiliki selisih yang besar dan menjadikan nilai *throughput* dari modulasi 32-QAM dengan *network coding* tertinggi. Namun dari hasil perbandingan tersebut, terlihat jelas bahwa *network coding* pada modulasi 16-QAM masih memberikan perbaikan *throughput* yang lebih besar dibandingkan pada level konstelasi lainnya.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Setelah dilakukan serangkaian tinjauan pustaka, perencanaan, simulasi, implementasi, pengukuran, dan penganalisan sistem, maka dapat ditarik kesimpulan sebagai berikut:

- Pada hasil simulasi dan implementasi nilai BER yang dihasilkan skema *Network Coding* lebih rendah dibandingkan dengan skema tanpa *Network Coding* namun besar selisih yang dihasilkan tidak signifikan sehingga masih dapat ditoleransi.
- Pada hasil simulasi semakin tinggi level konstelasi modulasi memberikan penurunan nilai BER yang semakin besar namun semakin tinggi level konstelasi modulasi tidak memberikan pengaruh terhadap nilai *Throughput*.
- Pada hasil implementasi semakin tinggi level konstelasi modulasi memberikan penurunan nilai BER yang semakin besar namun semakin tinggi level konstelasi modulasi memberikan kenaikan nilai *Throughput*.
- Pada kondisi *Outdoor* skema *Network Coding* memberikan hasil nilai BER dan nilai *Throughput* yang lebih tinggi dibandingkan pada kondisi *Indoor*.
- Persentase perbaikan nilai *Throughput* yang dihasilkan skema *Network Coding* terus meningkat mengikuti semakin besar level konstelasi yang digunakan hingga mencapai 16-QAM. Setelah itu, pada level konstelasi 32-QAM dan 64-QAM persentase perbaikan nilai *Throughput* mengalami penurunan.
- Perbaikan nilai *Throughput* yang dihasilkan pada hasil simulasi sebesar 31,8 % sedangkan pada hasil implementasi hanya mencapai nilai tertinggi sebesar 19,67 % pada modulasi 16-QAM pada kondisi *Outdoor*.
- Semakin besar jarak antar *Node* maka semakin besar penurunan nilai BER dan nilai *Throughput*.
- Pada hasil pengukuran level konstelasi 8-QAM, 16-QAM, dan 32-QAM, pada jarak 16 meter menghasilkan nilai BER dan nilai *Throughput* yang tinggi dibandingkan pada jarak yang lainnya. Sedangkan pada level konstelasi 4-QAM dan 64-QAM, hasil

pengukuran jarak 4 meter menghasilkan nilai BER dan nilai *Throughput* yang tinggi dibandingkan pada jarak yang lainnya.

## 5.2 Saran

Saran yang bisa diajukan sebagai bahan pengembangan atau penelitian lebih lanjut diantaranya:

- Penggunaan *Error Control Coding* tertentu pada sistem, guna meningkatkan kinerja sehingga memiliki *Probability of Error* yang lebih baik.
- Penggunaan skema *Network Coding* dengan metode XOR yang berbeda-beda.
- Penggunaan skema *Physical Network Coding* guna meningkatkan perbaikan nilai *Throughput*.



## BAB V PENUTUP

### 5.1 Kesimpulan

Setelah dilakukan serangkaian tinjauan pustaka, perencanaan, simulasi, implementasi, pengukuran, dan penganalisisan sistem, maka dapat ditarik kesimpulan sebagai berikut:

- Pada hasil simulasi dan implementasi nilai BER yang dihasilkan skema *Network Coding* lebih rendah dibandingkan dengan skema tanpa *Network Coding* namun besar selisih yang dihasilkan tidak signifikan sehingga masih dapat ditoleransi.
- Pada hasil simulasi semakin tinggi level konstelasi modulasi memberikan penurunan nilai BER yang semakin besar namun semakin tinggi level konstelasi modulasi tidak memberikan pengaruh terhadap nilai *Throughput*.
- Pada hasil implementasi semakin tinggi level konstelasi modulasi memberikan penurunan nilai BER yang semakin besar namun semakin tinggi level konstelasi modulasi memberikan kenaikan nilai *Throughput*.
- Pada kondisi *Outdoor* skema *Network Coding* memberikan hasil nilai BER dan nilai *Throughput* yang lebih tinggi dibandingkan pada kondisi *Indoor*.
- Persentase perbaikan nilai *Throughput* yang dihasilkan skema *Network Coding* terus meningkat mengikuti semakin besar level konstelasi yang digunakan hingga mencapai 16-QAM. Setelah itu, pada level konstelasi 32-QAM dan 64-QAM persentase perbaikan nilai *Throughput* mengalami penurunan.
- Perbaikan nilai *Throughput* yang dihasilkan pada hasil simulasi sebesar 31,8 % sedangkan pada hasil implementasi hanya mencapai nilai tertinggi sebesar 19,67 % pada modulasi 16-QAM pada kondisi *Outdoor*.
- Semakin besar jarak antar *Node* maka semakin besar penurunan nilai BER dan nilai *Throughput*.
- Pada hasil pengukuran level konstelasi 8-QAM, 16-QAM, dan 32-QAM, pada jarak 16 meter menghasilkan nilai BER dan nilai *Throughput* yang tinggi dibandingkan pada jarak yang lainnya. Sedangkan pada level konstelasi 4-QAM dan 64-QAM, hasil

pengukuran jarak 4 meter menghasilkan nilai BER dan nilai *Throughput* yang tinggi dibandingkan pada jarak yang lainnya.

## 5.2 Saran

Saran yang bisa diajukan sebagai bahan pengembangan atau penelitian lebih lanjut diantaranya:

- Penggunaan *Error Control Coding* tertentu pada sistem, guna meningkatkan kinerja sehingga memiliki *Probability of Error* yang lebih baik.
- Penggunaan skema *Network Coding* dengan metode XOR yang berbeda-beda.
- Penggunaan skema *Physical Network Coding* guna meningkatkan perbaikan nilai *Throughput*.

## DAFTAR PUSTAKA

- [1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.
- [2] M. Schwartz, *Information Transmission, Modulation, and Noise*, 3rd ed. McGraw-Hill, Inc, 1980.
- [3] P. Murphy, A. Sabharwal, and B. Aazhang, "Design of WARP: A wireless open-access research platform," in *Signal Processing Conference, 2006 14th European*, 2006, pp. 1–5.
- [4] F. G. Stremler, *Introduction to Communication System*. Addison-Wesley Publishing Company, 1982.
- [5] K. K. S. D, and S. G, *Quadrature Amplitude Modulation : A Simulation Study*. Canada: Simon Frasier University, 2005.
- [6] J. G. Proakis, *Digital Communications*. McGraw-Hill, 2001.
- [7] S. C. Liew, S. Zhang, and L. Lu, "Physical-layer network coding: Tutorial, survey, and beyond," *Phys. Commun.*, vol. 6, pp. 4–42, Mar. 2013.
- [8] Z. Wu, G. Xiang, Y. Fan, W. Bin, Y. Li, and Z. Luo, "A Physical-layer Network Coding Relay scheme for IEEE 802.11." 11-Jan-2013.
- [9] Suwadi and T. Suryani, *Sistem Komunikasi Nirkabel dan Aplikasinya pada WARP (Wirless Open Access Research Platform)*. 2014.
- [10] "WARPLab6/Architecture – WARP Project." [Online]. Available: <https://warpproject.org/trac/wiki/WARPLab6/Architecture>. [Accessed: 07-Dec-2015].
- [11] "WARP Project." [Online]. Available: <https://warpproject.org/trac>. [Accessed: 17-Nov-2015].
- [12] D. H. Woldegebreal and H. Karl, "Network-Coding-based Adaptive Decode and Forward Cooperative Transmission in a Wireless Network: Outage Analysis," *Eur. Wirel. Conf.*, Apr. 2007.
- [13] "WARPLab6/Specifications – WARP Project." [Online]. Available: <http://warpproject.org/trac/wiki/WARPLab6/Specifications>. [Accessed: 05-Dec-2015].



*Halaman ini sengaja dikosongkan*



## LAMPIRAN A

### LEMBAR PENGESAHAN PROPOSAL TA

Jurusan Teknik Elektro  
Fakultas Teknik Industri – ITS

**TU14159 TUGAS AKHIR – 4 SKS**

Nama Mahasiswa : Prastyanto  
Nomor Pokok : 2213 106 003  
Bidang Studi : Teknik Telekomunikasi dan Multimedia  
Tugas Diberikan : Semester Gasal 2015/2016  
Dosen Pembimbing : 1. Dr. Ir. Suwadi, M.T.  
2. Dr. Ir. Wirawan, DEA.

16 SEP 2015

Judul Tugas Akhir : **Implementasi dan Analisis Kinerja Network Coding pada Sistem Komunikasi Nirkabel Modulasi QAM Menggunakan WARP**  
(*Implementation and Performance Analysis Network Coding In QAM Modulation Wireless Communication Using WARP*)

**Uraian Tugas Akhir :**

Sistem komunikasi nirkabel yang merupakan bagian dari teknologi telekomunikasi kini menjadi solusi bagi setiap individu yang memiliki mobilitas yang cukup tinggi. Kebutuhan akan akses data yang cepat namun tidak berbenturan lurus dengan ketersediaan *bandwidth* yang terbatas dan *throughput* yang relatif rendah. Berdasarkan teori yang sudah ada mengenai Network Coding dan Modulasi Quadrature Amplitude Modulation (QAM), dibuat sebuah simulasi dengan konsep Sistem Komunikasi Nirkabel Modulasi QAM dengan Network Coding. Simulasi tersebut bertujuan untuk membuktikan konsep yang diteliti dapat direalisasikan dan sebagai pembuktian bahwa Network Coding dapat diaplikasikan pada Lapisan Aplikasi. Hasil simulasi diimplementasikan pada platform *Wireless Open-Access Research Platform* (WARP) yang merupakan salah satu jenis SDR (*Software Defined Radio*), memungkinkan untuk pengaplikasian dari konsep Sistem Komunikasi Nirkabel Modulasi QAM dengan Network Coding. Modul WARP tersebut akan digunakan sebagai model pemancar dan penerima untuk pengukuran kualitas kerja sistem Modulasi QAM dengan Network Coding pada suatu lingkungan *indoor* dan *outdoor* dengan berbagai jenis kondisi seperti perbedaan jarak dan *obstacle*. Dari hasil pengukuran tersebut diharapkan dapat membuktikan bahwa Network Coding dapat diimplementasikan pada modulasi QAM dengan meningkatnya *throughput* dan *bandwidth* menjadi lebih hemat hingga 10%.

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Ir. Suwadi, M.T.  
NIP. 19680618 1993031 002

Dr. Ir. Wirawan, DEA.  
NIP. 19631109 1989031 011

Mengetahui,  
Jurusan Teknik Elektro FT-ITS  
Ketua,

Mengetahui,  
Bidang Studi Telekomunikasi Multimedia  
Koordinator,

Dr. Tri Arief Sardjono, S.T., M.T.  
NIP. 19700212 1995121 001

Dr. Ir. Endrosono, DEA.  
NIP. 19650404 1991021 001

## LAMPIRAN B

### LISTING PROGRAM

#### 1. Program Utama

```

clc;
clear all;
close all;
warning off;

Pengukuran = false;

% Inisialisasi
channel = 6;
TxGainRF = [0:6:60 63];

Frame_Tx_1;
Frame_Tx_2;

Tx_Signal = n_sym_dat_s1;

for i = 1:length(TxGainRF)
    fprintf('\nTx Gain RF = %d dB \n',TxGainRF(i));
    %%
    #####
    ##### %%
    %           QAM Traditional           %
    %
    #####
    ##### %
    fprintf('QAM Traditional \n');
    fprintf('Timeslot 1 \n');
    if (Pengukuran)
        [rx_IQ, rx_RSSI] = WARP1_Transmit(TxGainRF(i),channel,xt_mf_s1);
    else
        rx_IQ = awgn(xt_mf_s1,TxGainRF(i));
    end

```

```

Frame_Rx_1;
Rx_Tra = yt_syms1;
S2_QAM_Tr = simbol_s1;
RSSI_Tr1 = rx_RSSI;

fprintf('Timeslot 2 \n');
if (Pengukuran)
    [rx_IQ, rx_RSSI] =
WARP2_Transmit(TxGainRF(i),channel,xt_mf_s2);
else
    rx_IQ = awgn(xt_mf_s2,TxGainRF(i));
end
Frame_Rx_2;
S1_QAM_Tr = simbol_s2;
RSSI_Tr2 = rx_RSSI;

%%
#####
##### %%
% QAM Relay %
%
#####
##### %
fprintf('\nQAM Relay \n');
fprintf('Timeslot 1 \n');
trelay = [];
tic;
if (Pengukuran)
    [rx_IQ, rx_RSSI] =
WARP3_Transmit(TxGainRF(i),channel,xt_mf_s1);
else
    rx_IQ = awgn(xt_mf_s1,TxGainRF(i));
end
Frame_Rx_1;
Sr1_QAM_Relay = simbol_s1;
RSSI_Relay1 = rx_RSSI;

fprintf('Timeslot 2 \n');

```

```

Relay_1;
if (Pengukuran)
    [rx_IQ, rx_RSSI] =
    WARP4_Transmit(TxGainRF(i),channel,xt_mf_relay);
else
    rx_IQ = awgn(xt_mf_relay,TxGainRF(i));
end
Frame_Rx_1;
Rx_Relay = yt_syms1;
S2_QAM_Relay = simbol_s1;
RSSI_Relay2 = rx_RSSI;

fprintf('Timeslot 3 \n');
if (Pengukuran)
    [rx_IQ, rx_RSSI] =
    WARP5_Transmit(TxGainRF(i),channel,xt_mf_s2);
else
    rx_IQ = awgn(xt_mf_s2,TxGainRF(i));
end
Frame_Rx_2;
Sr2_QAM_relay = simbol_s2;
RSSI_Relay3 = rx_RSSI;

fprintf('Timeslot 4 \n');
Mode = 6;
Relay_2;
if (Pengukuran)
    [rx_IQ, rx_RSSI] =
    WARP6_Transmit(TxGainRF(i),channel,xt_mf_relay);
else
    rx_IQ = awgn(xt_mf_relay,TxGainRF(i));
end
Frame_Rx_2;
S1_QAM_Relay = simbol_s2;
RSSI_Relay4 = rx_RSSI;
trelay = [trelay toc];

```



```

%%
#####
##### %%
%           QAM NC           %
%
#####
##### %
fprintf('\nQAM Network Coding \n');
fprintf('Timeslot 1 \n');
tnc = [];
tic;
if (Pengukuran)
    [rx_IQ, rx_RSSI] =
WARP3_Transmit(TxGainRF(i),channel,xt_mf_s1);
else
    rx_IQ = awgn(xt_mf_s1,TxGainRF(i));
end
Frame_Rx_1;
Sr1_QAM_NC = simbol_s1;
RSSI_NC1 = rx_RSSI;

fprintf('Timeslot 2 \n');
if (Pengukuran)
    [rx_IQ, rx_RSSI] =
WARP5_Transmit(TxGainRF(i),channel,xt_mf_s2);
else
    rx_IQ = awgn(xt_mf_s2,TxGainRF(i));
end
Frame_Rx_2;
Sr2_QAM_NC = simbol_s2;
RSSI_NC2 = rx_RSSI;

fprintf('Timeslot 3 \n');
Mode = 7;
Network_Coding;
if (Pengukuran)
    [rx_IQ, rx_RSSI] =
WARP7_Transmit(TxGainRF(i),channel,xt_mf_nc);
else

```

```

    rx_IQ = awgn(xt_mf_nc,TxGainRF(i));
end
Frame_Rx;
Rx_NC = yt_syms;
Sb = simbol;
RSSI_NC3 = rx_RSSI;

S2_QAM_NC = xor (simbol,simbol_b);
S1_QAM_NC = xor (simbol,simbol_a);
tnc = [tnc toc];

%% BER
% QAM Traditional
[N1,B1]=biterr(simbol_a,S2_QAM_Tr);
BER1(1,i)=B1;
Number1(1,i) = N1;

% QAM Relay
[N2,B2]=biterr(simbol_a,S2_QAM_Relay);
BER2(1,i)=B2;
Number2(1,i) = N2;
thr2 = (length(simbol_a)-Number2) ./ tnc;

% QAM NC
[N3,B3]=biterr(simbol_a,S2_QAM_NC);
BER3(1,i)=B3;
Number3(1,i) = N3;
thr3 = (length(simbol_a)-Number3) ./ tnc;

% Convert gain setting to Tx RF output in dBm according to MAX-2828
tx_gain_set = [8;16;24;32;40;48;56;63];
eqv_gain    = [-31;-28;-23;-19;-14.5;-11.5;-7;-2.5];
tx_outp     = interp1(tx_gain_set,eqv_gain,TxGainRF,'linear','extrap');
end

```

```

%%
#####
##### %%%
%           HASIL           %
%
#####
##### %
% BER vs Power Tx
a = figure(1);
semilogy(tx_outp, BER1, 'ko-', 'linewidth', 1); hold on;
semilogy(tx_outp, BER2, 'k+-', 'linewidth', 1); hold on;
semilogy(tx_outp, BER3, 'k*-', 'linewidth', 1); hold off;
grid on;
title('Kurva BER ');
xlabel('Tx Power Output (dBm)');
ylabel('BER');
legend('QAM Traditional', 'QAM Relay', 'QAM NC');

b = figure(2);
semilogy(tx_outp, thr2, 'ko-', 'linewidth', 1); hold on;
semilogy(tx_outp, thr3, 'k+-', 'linewidth', 1); hold off;
grid on;
title('Kurva Throughput');
xlabel('Tx Power Output (dBm)');
ylabel('Kbps');
legend('QAM Relay', 'QAM NC');

% Time PC
format shortg;
waktu = clock;
yy = num2str(waktu(1,1));
mm = num2str(waktu(1,2));
dd = num2str(waktu(1,3));
hh = num2str(waktu(1,4));
mn = num2str(waktu(1,5));

% Save File
nama_file = 'Data_pengukuran_4QAM_O30_LOS';

```

```
loc
'D:\1_Data_Pengukuran_Team\SemGanjil_2016\PRASTIYANTO\Resu
lt\Outdoor';
loc_fix = strcat (loc, '\Data_',dd,mm,yy, '_',hh, ':',mn, '\');
mkdir (loc_fix);
save([loc_fix nama_file '.mat']);
```

```
nama_file1 = '1Kurva_4QAM_O30_LOS';
nama_file2 = '2Kurva_4QAM_O30_LOS';
saveas(a,[loc_fix nama_file1 '.fig']);
print(a,'-dbitmap',[loc_fix nama_file1 '.bmp']);
saveas(b,[loc_fix nama_file2 '.fig']);
print(b,'-dbitmap',[loc_fix nama_file2 '.bmp']);
```

## 2. Program Pengirim

```
Fs = 40e6;
Ts = 1/Fs;

L = 2^14;
t = (0:L-1)/Fs;

M = 4;
k = log2(M);
bits = 20000;

a = [0 1];
simbol_a = randi (a,bits,1);

n_bits1 = length(simbol_a);
n_syms1 = n_bits1/k;

S1_tx_biner = reshape(simbol_a,log2(M),[]).';
S1_tx_desimal = bi2de (S1_tx_biner,2,'left-
msb');

h_bb_mod = modem.qammod('M',M,'PhaseOffset',0);
n_sym_dat_s1 = modulate(h_bb_mod,S1_tx_desimal);
```



```

Rsym = 3e6;

nsamp = floor(Fs/Rsym);
Rsym = Fs/nsamp;

pream = [0;0;0;0;-1+1i;1-1i;-1+1i;-1+1i;-1+1i;1-
1i;-1+1i;0;0;0;0;0;0;0;0];
pream_len = length(pream);

corr_win = 300;
pream_up = upsample(pream,nsamp);
ref_vect = zeros(corr_win,1);
ref_vect(1:length(pream_up)) = pream_up;
ref_matx =
toeplitz(ref_vect,circshift(ref_vect(corr_win:-
1:1),1));

txDelay = 400;

CSI_syms = 150;
CSI_dat = ones(CSI_syms,1);

shape = 'Square Root Raised Cosine';
N_Ord = 6;
R_SRRC = 0.9;
SRRC_spec =
fdesign.pulseshaping(nsamp,shape,'Nsym,Beta',N_O
rd,R_SRRC);
h_SRRC = design(SRRC_spec);
flt_delay = h_SRRC.order/2;
norm_factor = max(h_SRRC.numerator);
h_SRRC.numerator = h_SRRC.numerator/norm_factor;

data_sym = floor((L-(pream_len*nsamp)-txDelay-
flt_delay)/nsamp);

sym_size = floor(data_sym/(CSI_syms-1))-1;

```

```

sub_frame = CSI_syms-1;

fr_len = sub_frame*sym_size + CSI_syms;

data_sym = fr_len - CSI_syms;

n_frames = ceil(n_syms1/data_sym);
stuff_sym = [upsample(CSI_dat(1:CSI_syms-
1),sym_size+1);CSI_dat(end)];
n_dat_fr = repmat(stuff_sym,1,n_frames);
pad = n_frames*data_sym - n_syms1;

n_sym_dat_s1 = [n_sym_dat_s1;zeros(pad,1)];

n_sym_dat_s1 =
reshape(n_sym_dat_s1,data_sym,n_frames);

for ii=1:n_frames
    n_dat_fr(:,ii) =
sym_stuffing(n_dat_fr(:,ii),n_sym_dat_s1(:,ii),s
ym_size);
end

temp_mat1 = repmat(pream,1,n_frames);
xt_dat = [temp_mat1;n_dat_fr];

xt_mf_bb =
filter(h_SRRC,[upsample(xt_dat,nsamp);zeros(flt
delay,n_frames)]);
xt_mf_s1 = zeros(size(xt_mf_bb));

time = [(0:length(xt_mf_s1)-1)/Fs].';
f = 5e6;
for ii=1:n_frames
    xt_mf_s1(:,ii) =
xt_mf_bb(:,ii).*exp(1i*2*pi*f*time);

```

```

        norm_fact = max( [ max(real(xt_mf_sl(:,ii)))
        , max(imag(xt_mf_sl(:,ii))) ] );

```

```

        xt_mf_sl(:,ii) = xt_mf_sl(:,ii)./norm_fact;
end

```

### 3. Program *Transmit*

```

function [rx_IQ] =
WARP1_Transmit(TxGainRF,channel,st)

USE_AGC = false;

NUMNODES = 3;
nodes = wl_initNodes(NUMNODES);

eth_trig = wl_trigger_eth_udp_broadcast;
wl_triggerManagerCmd(nodes,'add_ethernet_trigger
',[eth_trig]);

[RFA,RFB] = wl_getInterfaceIDs(nodes(1));

wl_interfaceCmd(nodes,'RF_ALL','tx_gains',3,TxGa
inRF);
wl_interfaceCmd(nodes,'RF_ALL','channel',2.4,cha
nnel);

if(USE_AGC)

wl_interfaceCmd(nodes,'RF_ALL','rx_gain_mode','a
utomatic');
    wl_basebandCmd(nodes,'agc_target',-10);
    wl_basebandCmd(nodes,'agc_trig_delay', 511);
else

wl_interfaceCmd(nodes,'RF_ALL','rx_gain_mode','m
anual');

wl_interfaceCmd(nodes,'RF_ALL','rx_gains',1,1);

```

end

```
txLength = nodes(1).baseband.txIQLen;

wl_basebandCmd(nodes, 'tx_delay', 0);
wl_basebandCmd(nodes, 'tx_length', txLength);

node_s1 = nodes(1);
node_sr = nodes(2);
node_s2 = nodes(3);

RF_TX = RFA;
RF_RX = RFA;

Ts =
1/(wl_basebandCmd(nodes(1), 'tx_buff_clk_freq'));
Ts_RSSI =
1/(wl_basebandCmd(nodes(1), 'rx_rssi_clk_freq'));

st_size = size(st);
rx_IQ = zeros(txLength, st_size(2));
rx_RSSI = zeros(txLength/4, st_size(2));
rx_gains = zeros(3, st_size(2));

wl_interfaceCmd(node_s1, RF_TX, 'tx_en');
wl_interfaceCmd(node_s2, RF_RX, 'rx_en');
wl_basebandCmd(node_s1, RF_TX, 'tx_buff_en');
wl_basebandCmd(node_s2, RF_RX, 'rx_buff_en');

tic;

for m=1:st_size(2);
    wl_basebandCmd(node_s1, [RF_TX], 'write_IQ',
    st(:,m) );
    eth_trig.send();
```



```

        rx_IQ(:,m) =
wl_basebandCmd(node_s2,[RF_RX],'read_IQ', 0,
txLength);
        rx_RSSI(:,m) =
wl_basebandCmd(node_s2,[RF_RX],'read_RSSI',0,txL
ength/(Ts_RSSI/Ts));
        rx_gains(:,m) =
wl_basebandCmd(node_s2,[RF_RX],'agc_state');
end
toc;

wl_basebandCmd(nodes,'RF_ALL','tx_rx_buff_dis');
wl_interfaceCmd(nodes,'RF_ALL','tx_rx_dis');

```

#### 4. Program Penerima

```

h_bb_demod =
modem.qamdemod('M',M,'PhaseOffset',0);

time = ((0:length(rx_IQ)-1)/Fs).';
rx_IQ_disp = rx_IQ;
for ii=1:n_frames
    vco_sig = exp(1i*2*pi*f*time);
    rx_IQ(:,ii) = rx_IQ(:,ii).*conj(vco_sig);
end

hrx_SRRC = design(SRRC_spec);
hrx_SRRC.numerator = hrx_SRRC.numerator *
(norm_factor*nsamp);
yt_mf =
filter(hrx_SRRC,[rx_IQ;zeros(flt_delay,n_frames)
]);

yt_rx_data = zeros(size(n_dat_fr));
z_sym1 = zeros(size(n_sym_dat_s1));
ch_est = zeros(size(n_dat_fr));

for ii=1:n_frames

```

```

temp_vct1 = yt_mf(:,ii);

    correlation = abs( (temp_vct1(1:corr_win'))
* ref_matx );
    pream_start = find(correlation ==
max(correlation));
    first_sample = pream_start+length(pream_up);

    temp_vct2 = temp_vct1(first_sample:end);
    temp_vct3 =
downsample(temp_vct2(1:end),nsamp);

    temp_vct3 = temp_vct3(1:fr_len);
    yt_rx_data(:,ii) = temp_vct3;

    ch_est(:,ii) =
estimate_chan(yt_rx_data(:,ii),sym_size);

    temp_vct3 =
yt_rx_data(:,ii).*conj(ch_est(:,ii));
    multip = [abs(ch_est(:,ii)).^2];

    yt_rx_data(:,ii) = (temp_vct3)./multip;

    temp_vct3 =
sym_destuffing(yt_rx_data(:,ii),sym_size);

    z_sym1(:,ii) = temp_vct3;
end

yt_syms1 = reshape(z_sym1,data_sym*n_frames,1);
yt_syms1 = yt_syms1(1:length(S1_tx_desimal));

%Demodulasi
z_sym1 = demodulate(h_bb_demod,yt_syms1);

```

```

biner_s1 = de2bi(z_sym1,log2(M),'left-msb').';
simbol_s1 = reshape(biner_s1,1,[]).';

```

## 5. Simulasi

```

clc;
clear all;
close all;

M = 64;
% M-QAM
nsimbol = 30000;
% Jumlah bit yang dikirim
k = log2(M);

%% Pembangkitan data

% Data a
a = [0 1];
simbol_a = randi (a,nsimbol,1);

% Data b
b = [0 1];
simbol_b = randi (b,nsimbol,1);

snr = 0:50;
% Nilai SNR
for x = 1:51

%% Kanal 1 (dengan dan tanpa Network Coding)

% Bagian Pengirim Node S1 ke Node SR
S1_tx_biner = reshape(simbol_a,k,[]).';
% Konversi serial bit ke paralel bit
S1_tx_desimal = bi2de (S1_tx_biner,2,'left-
msb'); % Konversi bit
ke desimal

```

```

S1_tx = qammod(S1_tx_desimal,M);
% Modulasi QAM Node S1

% Bagian Penrima Relay Node SR dari Node S1
S1_tx_noise = awgn (S1_tx,snr(x), 'measured');
% Pemberian noise AWGN pada kanal 1

%% Kanal 2 (tanpa Network Coding)

% Bagian Relay Node SR ke Node S2
SR_S1_noise = awgn
(S1_tx_noise,snr(x), 'measured');

% Bagian Penerima Node S2 dari Node SR
S1 = qamdemod (SR_S1_noise,M);
S1_biner = de2bi(S1,k, 'left-msb').';
simbol_s1 = reshape(S1_biner,1,[]).';

%% Kanal 3 (dengan dan tanpa Network Coding)

% Bagian Pengirim Node S2 ke Node SR
S2_tx_biner = reshape(simbol_b,k,[]).';
% Konversi serial bit ke paralel bit
S2_tx_desimal = bi2de (S2_tx_biner,2, 'left-
msb'); % Konversi bit
ke desimal
S2_tx = qammod(S2_tx_desimal,M);
% Modulasi QAM Node S2

% Bagian Penrima Relay Node SR dari Node S2
S2_tx_noise = awgn (S2_tx,snr(x), 'measured');
% Pemberian noise AWGN pada kanal 2

%% Kanal 4

% Tanpa Network Coding
% Bagian Relay Node SR ke Node S2

```



```

SR_S2_noise = awgn
(S2_tx_noise,snr(x),'measured');

% Bagian Penerima Node S2 dari Node SR
S2 = qamdemod (SR_S2_noise,M);
S2_biner = de2bi(S2,k,'left-msb').';
simbol_s2 = reshape(S2_biner,1,[]).';

% Dengan Network Coding
% Bagian Relay dengan Network Coding Node SR
- S1 dan S2
S1_demod = qamdemod (S1_tx_noise,M);
S1_demod_biner = de2bi (S1_demod,k,'left-
msb').';
S1_demod_bit =
reshape(S1_demod_biner,1,[]).';

S2_demod = qamdemod (S2_tx_noise,M);
S2_demod_biner = de2bi (S2_demod,k,'left-
msb').';
S2_demod_bit =
reshape(S2_demod_biner,1,[]).';

Sr = xor(S1_demod_bit,S2_demod_bit);
Sr_biner = reshape(Sr,k,[]).';
Sr_desimal = bi2de (Sr_biner,2,'left-msb');
Sr_mod = qammod(Sr_desimal,M);

Sr_noise = awgn (Sr_mod,snr(x),'measured');

Sr_rx = qamdemod (Sr_noise,M);
% Demodulasi QAM Node S1
Sr_rx_biner = de2bi(Sr_rx,k,'left-msb').';
% Konversi desimal ke bit
Sr_bit = reshape(Sr_rx_biner,1,[]).';
% Konversi paralel bit ke serial bit

% Bagian Penerima Node S1 dari Node SR

```

```

data_s2 = xor(Sr_bit,simbol_a);

% Bagian Penerima Node S2 dari Node SR
data_s1 = xor(Sr_bit,simbol_b);

%% Bagian BER

% BER QAM Traditional
[N1,B1]=biterr(simbol_a,simbol_s1);
BER1(1,x)=B1;
Number1(1,x) = N1;

thr1 = (length(simbol_a)-Number1) ./
(1*4*length(simbol_a));

% BER QAM NC
[N2,B2]=biterr(simbol_a,data_s1);
BER2(1,x)=B2;
Number2(1,x) = N2;

thr2 = (length(simbol_a)-Number2) ./
(1*3*length(simbol_a)); % SNC
% thr3 = (length(simbol_a)-Number2) ./ (1.25e-
6*2*length(simbol_a)); % PNC
end

%% Hasil Simulasi

% BER
figure (1);
semilogy(snr,BER1,'k-','linewidth',2); hold on;
semilogy(snr,BER2,'ko-','linewidth',2); hold
off;
grid on;
title('Simulasi BER pada Modulasi 64-QAM');
xlabel('SNR (dB)');
ylabel('BER');
legend('QAM Non NC','QAM NC');

```

```

% Throughput
figure (2);
plot(snr,thr1,'k*-','linewidth',2); hold on;
plot(snr,thr2,'ko-','linewidth',2); hold on;
% semilogy(snr,thr3,'k+-','linewidth',2); hold
off;
grid on;
title('Simulasi Throughput pada Modulasi 64-
QAM');
xlabel('SNR');
ylabel('Throughput (bps)');
legend('QAM Non NC','QAM SNC','QAM PNC');

% Plot Konstelasi QAM Tradisional
h = scatterplot(SR_S2_noise(1,10),1,0,'g*');
hold on;
scatterplot(S1_tx(1,10),1,0,'k.',h); hold off;
grid on;
title('Constellation QAM Traditional');
legend('Received','Transmitted');
axis([-2 2 -2 2]);
%
% % Plot Konstelasi QAM NC
% h = scatterplot(Sr_noise,1,0,'g*'); hold on;
% scatterplot(S1_tx,1,0,'k.',h); hold off;
% grid on;
% title('Constellation QAM Traditional');
% axis([-2 2 -2 2]);
% legend('Received','Transmitted');

```

## LAMPIRAN C DOKUMENTASI

1. Pengukuran dalam ruangan, di AJ 403, Teknik elektro, ITS



2. Pengukuran luar ruangan, di Area Parkir Teknik Elektro, ITS





## BIODATA PENULIS



Penulis bernama lengkap Prastiyanto, anak pertama dari dua bersaudara yang lahir di Jakarta pada tanggal 6 Juni 1992. Penulis memulai pendidikan tingkat SD pada tahun 1998 di SDN 07 Sunter Agung dan pindah saat melanjutkan ke kelas 5 di SDN SSN 13 Sunter Agung, Jakarta. Lulus dari SMPN 116 Jakarta pada tahun 2007, dan SMK Telkom Sandhy Putra Jakarta pada tahun 2010. Gelar Ahli Madya (D3) diperoleh pada tahun 2013 dari Program Studi Teknik Telekomunikasi, Jurusan Teknik Elektro, Politeknik Negeri Jakarta, Depok. Pada tahun yang sama penulis melanjutkan tingkat Sarjana di Bidang Studi Teknik Telekomunikasi dan Multimedia, Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember, Surabaya. Selama menempuh jenjang Ahli Madya, penulis aktif di organisasi Himpunan Mahasiswa Elektro PNJ. Penulis dapat dihubungi melalui alamat email: [tiyan.prastiyanto@gmail.com](mailto:tiyan.prastiyanto@gmail.com).